

T.M.
(043)62
2020
D380

TESIS DE MAESTRÍA EN INGENIERÍA

UN ENFOQUE DE APRENDIZAJE PROFUNDO PARA LA CUANTIFICACIÓN DE LA FUNCIÓN CARDIACA

Lucca Dellazoppa
Maestrando

Dr. Ariel H. Curiale
Director

Dr. Germán Mato
Co-director

Miembros del Jurado

Dra. Inés Samengo (Instituto Balseiro)
Dr. Matías Calandrelli (Sanatorio San Carlos de Bariloche)
Dr. Sebastián Risau (Instituto Balseiro)

18 de Diciembre de 2020

Departamento de Física Médica – Centro Atómico Bariloche

Instituto Balseiro
Universidad Nacional de Cuyo
Comisión Nacional de Energía Atómica
Argentina

INVENTARIO: 24189
21.09.21
Biblioteca Leo Falicov

Índice de símbolos

Símbolo	Significado
AE	Autoencoder
AI	Inteligencia artificial
BD	Base de datos
BSA	Índice de superficie corporal
CNN	Red neuronal convolucional
CT	Tomografía computada
DL	<i>Deep Learning</i>
ED	Telediástole
EDV	Volumen ventricular en telediástole
$EDV_{\text{myo}}^{(LV)}$	Volumen del Myo en telediástole
EF	Fracción de eyección
ES	Telesístole
ESV	Volumen ventricular en telesístole
GJD	Distancia de <i>Jaccard</i> generalizada
GUI	Interfaz gráfica de usuario
HD	Distancia de <i>Hausdorff</i>
LV	Ventrículo izquierdo
LVEF	Fracción de eyección correspondiente al LV
LVM	Masa del miocardio correspondiente al LV
LVV	Volumen ventricular izquierdo
ML	<i>Machine Learning</i>
MR	Resonancia magnética nuclear
MRML	Modelo de datos <i>Medical Reality Modeling Language</i>
MVC	Patrón de diseño <i>Modelo Vista Controlador</i>
Myo	Miocardio correspondiente al LV
R	Coefficiente de <i>Pearson</i>
RL	<i>Representation Learning</i>
RNA	Red neuronal artificial
ROI	Región de interés en una imagen
RV	Ventrículo derecho
RVEF	Fracción de eyección correspondiente al RV
RVV	Volumen ventricular derecho
SAX MR-C	Estudios del ciclo cardiaco (<i>Short Axis MR Cine</i>)
SV	Volumen eyectado
US	Ultrasonido
VAE	Autoencoder variacional
D_{KL}	Divergencia de <i>Kullback-Leibler</i>
\mathbb{E}	Error de reconstrucción de un VAE
\mathcal{L}	Cota inferior o <i>variational lower bound</i>
p_{θ}	Decodificador de un AE
$p_{\theta}(\mathbf{x}^{(i)})$	<i>Likelihood</i> marginal de la reconstrucción $\mathbf{x}^{(i)}$ en un VAE
ρ_{myo}	Densidad del tejido del miocardio
q_{ϕ}	Codificador de un AE

Tabla 1: Abreviaturas y símbolos utilizados en orden alfabético.

Resumen

Las enfermedades cardiovasculares son la primera causa de muerte a nivel mundial. Es por esto que las tareas de detección temprana, diagnóstico y pronóstico resultan de vital importancia para las personas que padecen o corren riesgo de padecer estas patologías. Dos indicadores ampliamente utilizados para cuantificar la función y estructura cardíaca son la fracción de eyección (EF) y la masa del miocardio correspondiente al ventrículo izquierdo (LVM), cuyo cálculo requiere identificar y delinear de forma precisa los contornos de distintos tejidos del corazón en estudios de imágenes médicas. Si bien existen métodos semiautomáticos para realizar esta tarea, no suelen tener la precisión alcanzada mediante la demarcación manual, la cual es comúnmente considerada tediosa, repetitiva y puede demandar alrededor de 20 minutos por paciente para un médico experto. En este contexto, se propone trasladar al ámbito clínico una aplicación que permita cuantificar la función cardíaca a partir de la estimación de la EF y LVM de manera automática en estudios de resonancia magnética nuclear; con precisiones y errores comparables a los obtenidos por especialistas. Con ese objetivo se diseñó y desarrolló la herramienta de código abierto CardIAc, la cual incorpora dos modelos de redes neuronales profundas que se entrenaron para detectar las estructuras cardíacas de interés. De esta manera, se presenta un enfoque automático que logra reducir el tiempo de los métodos tradicionales a 2 min. (CPU) y 30 seg. (GPU), con errores en la cuantificación de EF (0.02 ± 4.6 [%]) que se encuentran en el rango inter-observador actual. Asimismo, este enfoque contempla la posibilidad de realizar una corrección manual de los tejidos identificados en la etapa de detección de estructuras. La herramienta provista se desarrolló como una extensión del *software* de visualización y procesamiento de imágenes médicas 3DSlicer, integrándose así con el resto de utilidades nativas.

Palabras clave: Código Abierto, Cuantificación Cardíaca, Segmentación Cardíaca, Redes Neuronales Convolucionales

Abstract

Cardiovascular diseases are the leading cause of death worldwide. That is why the tasks of early detection, diagnosis and prognosis are of paramount importance for people who suffer or are at risk of suffering from these pathologies. Two descriptors widely used for the quantification of cardiac function and structure are the ejection fraction (EF) and the left ventricular myocardial mass (LVM), whose calculation requires identifying and precisely delineating the contours of different heart tissues in medical imaging-based studies. Although there are semi-automatic methods to perform this task, they usually do not achieve the precision obtained through manual demarcation, which is commonly considered tedious and repetitive, requiring up to 20 minutes per patient for an expert. In this context, we aim to provide the clinical field with an application that allows the automatic quantification of cardiac function (EF and LVM) in nuclear magnetic resonance imaging studies, with accuracies and errors similar to those obtained by specialists. With this objective, the open source tool CardIAc was designed and developed, which incorporates two models of deep neural networks that were trained to identify the cardiac structures of interest. In this way, we present an automatic approach that manages to reduce the time of traditional methods to 2 min. (CPU) and 30 sec. (GPU), with an observed error in EF quantification (0.02 ± 4.6 [%]) within the current inter-observer range. Besides, the capability of making manual corrections to the identified tissues is covered in this approach. The tool provided was developed as an extension of the 3DSlicer medical image visualization and processing software, thus integrating with the rest of native utilities.

Keywords: Open Source, Cardiac Quantification, Cardiac Segmentation, Convolutional Neural Networks

Índice de contenidos

Índice de símbolos	i
1. Introducción	1
1.1. Objetivos	3
1.2. Metodología	3
1.3. Contribuciones de la tesis	5
1.4. Estructura del trabajo	6
2. Marco Teórico	7
2.1. Cuantificación de la función cardiaca	7
2.2. Técnicas basadas en aprendizaje automático	11
2.3. <i>Software</i> 3DSlicer	18
3. CardIAc: una aplicación para la cuantificación cardiaca	20
4. Materiales y modelos de redes neuronales	26
4.1. Materiales	26
4.2. Modelos de redes neuronales propuestos	27
5. Resultados	31
5.1. Construcción de la ROI	32
5.2. Identificación de estructuras cardiacas	33
5.3. Cuantificación de la función cardiaca	37
5.4. Robustez de la cuantificación cardiaca	40
6. Conclusiones y trabajos futuros	42
Bibliografía	44

Capítulo 1

Introducción

A lo largo de las últimas dos décadas y en la actualidad, las enfermedades cardiovasculares se han posicionado como la primera causa de muerte a nivel mundial, con una tendencia creciente^[1,2]. En el año 2017 representaron el 31.8 % de las muertes mundiales, entre las que se destacan la enfermedad coronaria, hipertensión, enfermedad cerebrovascular e insuficiencia cardíaca. La segunda causa, ese mismo año, fue el cáncer con el 17.2 % de las muertes mundiales. El resto de decesos fueron debidos a causas varias, entre las que se destacan enfermedades respiratorias, demencia, diabetes y accidentes de tránsito. Cada una de estas últimas representa en sí misma menos del 7 % de las muertes anuales. Esto deja en evidencia el gran peso de las enfermedades cardíacas sobre el resto de causas en la mortalidad a nivel mundial.

En vista de esta realidad, las tareas de detección temprana, seguimiento y asistencia médica para las personas que padecen o se encuentran en el grupo de riesgo de enfermedades cardiovasculares resultan de vital importancia para mejorar su calidad de vida. Dos indicadores que cumplen un rol significativo en el diagnóstico y evaluación de patologías como insuficiencias cardíacas, infartos de miocardio y enfermedades de las válvulas cardíacas son la fracción de eyección (EF) y la masa del tejido del miocardio correspondiente al ventrículo izquierdo (LVM)^[3,4]. Ambos indicadores son ampliamente utilizados para la cuantificación de la función y estructura del corazón, siendo la EF considerada como la *piedra angular* para las directrices y prácticas de la cardiología moderna^[5].

Para el cálculo de estos indicadores es necesario identificar tres estructuras cardíacas: los ventrículos izquierdo (LV) y derecho (RV), y el miocardio correspondiente al ventrículo izquierdo (Myo). Luego, estimando los volúmenes de dichas estructuras es posible calcular la EF y LVM. Debido a que la EF es un indicador dinámico de la función del corazón, se necesita identificar el ventrículo en dos instantes del ciclo cardíaco denominados telediástole (ED) y telesístole (ES), correspondientes a los momentos con mayor y menor volumen de sangre en su interior. Este descriptor se utiliza tanto para el LV (LVEF) como para el RV (RVEF). Por otro lado, la LVM es un indicador estático de la estructura cardíaca y para su cálculo se precisa detectar el Myo en el instante de telediástole. De esta forma, detectar con precisión ambos ventrículos y el Myo es un paso fundamental para poder cuantificar correctamente la EF y LVM.

Actualmente, las técnicas no invasivas más utilizadas para cuantificar la función del

corazón son la resonancia magnética (MR), tomografía computada (CT) y ultrasonido (US). En este contexto, la MR presenta ventajas de resolución respecto al US ya que permite obtener imágenes más claras y precisas que facilitan la distinción entre los distintos tejidos. Además, no involucra el uso de radiaciones ionizantes perjudiciales para la salud, lo cual representa una clara ventaja respecto a la CT. Por lo tanto, es de esperar que la resonancia magnética cardíaca sea considerada actualmente como un *estándar de oro* en la detección de estructuras y cuantificación de la función cardíaca^[6,7].

Mediante estudios tipo SAX MR-C (*Short Axis MR Cine*), formados por imágenes de resonancia magnética del corazón a lo largo del ciclo cardíaco, es posible estimar los volúmenes ventriculares en ED y ES, y del miocardio en ED. Para ello, los especialistas necesitan identificar los contornos del endocardio (el tejido interno del miocardio) y del Myo sobre las distintas imágenes del estudio. Este proceso de identificación y clasificación de diversas estructuras es conocido como *segmentación* en el área de procesamiento de imágenes. Una vez identificados los contornos en telediástole y telesístole, pueden estimarse los volúmenes para derivar posteriormente la EF y LVM. En la práctica clínica, la delineación de los tejidos forma parte del flujo de trabajo típico para cuantificar dichos indicadores (ver Figura 1.1).

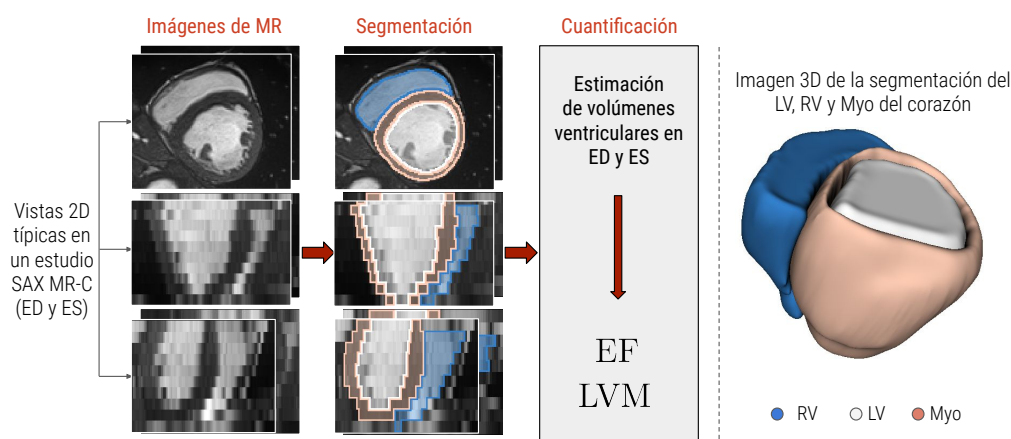


Figura 1.1: Flujo de trabajo típico en cardiología para cuantificar los indicadores de la función cardíaca a partir de un estudio de tipo SAX MR-C. Para mayor claridad anatómica, se muestra a la derecha una imagen 3D de la identificación del LV, RV y Myo del corazón.

Existen diversas herramientas informáticas de visualización y procesamiento de imágenes médicas que permiten realizar la identificación de los contornos y posterior cuantificación en imágenes médicas. Dos comúnmente utilizadas son *OsiriX*^[8] (disponible para *Mac*) y *Segment*^[9] (disponible para *Mac* y *Windows*). A pesar de las facilidades que ofrecen estas herramientas, la segmentación todavía suele considerarse una tarea tediosa y repetitiva, y requiere de gran intervención manual por parte del especialista. Por ejemplo, delinear los contornos para los instantes de ED y ES puede llevarle alrededor de 20 minutos por paciente a un experto^[10]. Estos tiempos son aún más extensos cuando se trata de todo el ciclo cardíaco, el cual puede involucrar el análisis de más de 15 imágenes volumétricas por paciente. En este sentido, resulta cada vez más frecuente el trabajo interdisciplinario entre la medicina y otras disciplinas como la ingeniería, física, matemática y ciencias de la computación, para proveer nuevas tecnologías diseñadas en base a las necesidades y desafíos reales enfrentados día a día en la clínica.

A raíz de esto, es posible plantearse las siguientes preguntas: ¿Será posible automatizar el proceso de segmentación con precisiones similares a las de un experto? ¿Existirá algún enfoque que facilite aún más a los médicos la realización de esta tarea repetitiva y que reduzca su tiempo? En caso afirmativo, ¿Asegurará errores dentro de los rangos cometidos actualmente por diferentes especialistas al clasificar y delinear las estructuras?

Desde la aparición y mejora de procesadores con alto poder de cómputo, las técnicas de aprendizaje profundo basadas en redes neuronales artificiales han ganado popularidad para resolver una gran cantidad de problemas. En el año 2015, se presentó el modelo denominado U-Net^[11], capaz de resolver la tarea de clasificación de estructuras en imágenes médicas de forma automática y con una precisión notable para la época. Ese mismo año fue ganador del primer premio en la competencia *ISBI cell tracking challenge 2015*, segmentando estructuras neuronales. A partir de entonces, diversos trabajos se han desarrollado mejorando y extendiendo el modelo U-Net para resolver el problema de identificación de tejidos y estructuras en imágenes cardiovasculares.

La motivación de este trabajo interdisciplinario de ingeniería radica en la idea de acercar al área de la cardiología una herramienta novedosa y desarrollada en código abierto, que permita automatizar el proceso de segmentación y cuantificación de la función cardiaca. En ese sentido, se desarrolló la herramienta llamada CardIAc, la cual permite cuantificar la EF y LVM con precisiones comparables a las obtenidas actualmente por diferentes especialistas.

1.1. Objetivos

El objetivo general de este trabajo es diseñar, desarrollar y trasladar al ámbito clínico una aplicación que permita cuantificar la función cardiaca de ambos ventrículos del corazón sobre estudios SAX MR-C. Para ello, se propone alcanzar los siguientes objetivos parciales:

- Realizar un estudio de los principales componentes involucrados en el proceso de cuantificación de la función cardiaca sobre estudios SAX MR-C, poniendo particular énfasis en los enfoques de aprendizaje profundo utilizados para identificar las estructuras cardiacas.
- Implementar un prototipo capaz de identificar los ventrículos y el músculo del corazón (LV, RV y Myo) de forma automática, mediante la incorporación de modelos de segmentación basados en técnicas de aprendizaje profundo.
- Desarrollar y proveer una aplicación simple y fácil de usar en la clínica con la que sea posible completar el proceso de cuantificación de la LVM y la EF de ambos ventrículos de forma automática, con errores dentro del rango inter observador cometido por los especialistas.

1.2. Metodología

La metodología propuesta para alcanzar los objetivos del trabajo se agrupa en cuatro etapas: el estudio del estado del arte, la preparación de las herramientas compu-

tacionales necesarias, la implementación de la aplicación y la validación de los métodos propuestos. A continuación se detalla cada una de ellas:

■ **Estudio del estado del arte**

- Estudio de los fundamentos de la anatomía y fisionomía del corazón, con especial énfasis en la morfología de las estructuras del LV, RV y Myo y su variación a lo largo del ciclo cardiaco.
- Estudio del procedimiento de cuantificación de la EF y la LVM a partir de la estimación de los volúmenes de diferentes estructuras cardíacas.
- Estudio del flujo de trabajo clínico involucrado en la cuantificación de la EF y la LVM, con especial énfasis en la cuantificación sobre estudios SAX MR-C.
- Estudio de los conceptos de aprendizaje automático, redes neuronales artificiales y aprendizaje profundo. Se estudiarán las arquitecturas más utilizadas para resolver problemas de identificación de estructuras en imágenes médicas, haciendo hincapié en la arquitectura U-Net. Además se analizarán diferentes enfoques de la bibliografía para la incorporación de información anatómica en los modelos.

■ **Preparación de las herramientas computacionales**

- Lenguaje de programación: Se estudiarán conceptos de programación aplicados al lenguaje *python*, el cual será utilizado en este trabajo para la implementación de los distintos algoritmos y prototipos.
- Estudio del *software* 3DSlicer: Este *software* de visualización y procesamiento de imágenes médicas de código abierto permite incorporar nuevas funcionalidades mediante el desarrollo de extensiones. Se estudiará su funcionamiento, el sistema de representación de datos internos (MRML) y la factibilidad de incorporar modelos de identificación de estructuras cardíacas basados en redes neuronales profundas dentro del mismo.
- Estudio de librerías: Para las diversas operaciones matemáticas y/o estadísticas y para las tareas que requieran el manejo de datos “n-dimensionales” se utilizarán las librerías *numpy* y *scipy*, especializadas en computación científica; mientras que se usará la librería *matplotlib* para verificar y visualizar los diferentes resultados intermedios que se vayan obteniendo. Para las tareas de visualización y procesamiento en imágenes médicas se considerarán las facilidades que brindan las librerías *SimpleITK* y *VTK*. Por otro lado, se estudiarán las librerías *tensorflow* y *keras* para la implementación de los prototipos capaces de identificar las estructuras cardíacas. Dichas librerías se especializan en la implementación y optimización de redes neuronales. Por último, para el desarrollo de extensiones e interfaces gráficas de usuario dentro del *software* 3DSlicer se estudiará la librería *qt*.

■ **Implementación de la aplicación**

- Realización de prototipos: Se implementarán primeras aproximaciones de algoritmos y extensiones que permitan identificar y visualizar estructuras

de forma automática dentro del entorno de 3DSlicer. A partir de estas, se irán incorporando funcionalidades de forma gradual, hasta lograr una herramienta que permita realizar el proceso de cuantificación de la función cardiaca de forma completa dentro del *software*.

- Verificación visual de los métodos. Para ello, se visualizarán los resultados intermedios mediante las facilidades de visualización de 3DSlicer y las librerías *qt* y *matplotlib*.
- Manual de uso: Una vez finalizada la aplicación, se proveerá de un manual de usuario con el detalle de su descarga, instalación, funcionamiento y desinstalación. El manual incorporará ejemplos de uso típico y las utilidades previstas para intervenir de forma manual en el caso que se obtengan errores con el método automático propuesto.

■ Validación de los métodos propuestos

- Análisis cualitativo y cuantitativo: Se analizarán los resultados del método de identificación de estructuras propuesto de forma cualitativa, observando la plausibilidad anatómica de las estructuras detectadas; y de forma cuantitativa, utilizando la información disponible en las bases de datos utilizadas para entrenar los modelos. Adicionalmente, se compararán los resultados y errores obtenidos al cuantificar los indicadores de la función cardiaca con los observados en la bibliografía y los obtenidos por especialistas.

1.3. Contribuciones de la tesis

El presente proyecto se incorporó desde sus inicios al grupo de investigación del Laboratorio de Procesamiento de Imágenes y Señales del Departamento de Física Médica (Instituto Balseiro), en la línea de Visión Artificial en imágenes médicas. De esta forma, se contó con un punto de partida dado por^[12,13]. Estos trabajos realizados de forma colaborativa, junto con los desarrollos de esta tesis dieron lugar a las siguientes publicaciones en congresos y revistas:

[1] A. H. Curiale, M. E. Calandrelli, L. Dellazoppa, M. Trevisan, J. L. Bocián, J. P. Bonifacio, G. Mato, “Cuantificación Automática de los Volúmenes y Función de Ambos Ventrículos en Resonancia Cardiaca: Propuesta y Evaluación de un Método de Inteligencia Artificial”, *Revista Argentina de Cardiología* (*under revision*).

[2] L. Dellazoppa, A. H. Curiale, and G. Mato, “Un enfoque de aprendizaje profundo para la cuantificación de la función cardiaca”, in *Workshop Machine Intelligence: Training and Applications to Regional Challenges*, Facultad de Ciencias Exactas y Tecnología, Universidad Nacional de Tucumán, 2020.

[3] A. H. Curiale, L. Dellazoppa and G. Mato, “CardIAc: Cuantificación de la Función Cardiaca mediante la Detección Automática del ventrículo Izquierdo en Imágenes de Resonancia Magnética Nuclear”, in *Taller Argentino de Computación Científica*, Facultad de Ciencias Físico Matemáticas y Naturales, Universidad Nacional de San Luis, 2019.

1.4. Estructura del trabajo

El presente documento se estructura de la siguiente forma: En el capítulo 2 se introducen las bases teóricas del trabajo, las cuales se dividen en tres secciones. En la sección 2.1 se resumen los conceptos útiles de anatomía, fisiología del corazón y cuantificación de la función cardíaca. En la sección 2.2 se describen los conceptos de redes neuronales artificiales, aprendizaje profundo y se presenta la arquitectura U-Net diseñada para resolver el problema de identificación de estructuras en imágenes médicas. Finalmente, en la sección 2.3 se presenta el *software* 3DSlicer, un entorno de visualización y procesamiento de imágenes médicas de código abierto, dentro del cual se desarrolló CardIAc. En el capítulo 3 se expone el funcionamiento e implementación de CardIAc, su utilidad e integración con los diferentes módulos del *software* 3DSlicer. En el capítulo 4 se detallan los modelos propuestos para identificar estructuras cardíacas y cuantificar los indicadores de interés de este trabajo. En el capítulo 5 se muestran los resultados obtenidos de segmentación y cuantificación sobre dos bases de datos diferentes. Por último, en el capítulo 6, se presentan las conclusiones y se proponen trabajos futuros.

Capítulo 2

Marco Teórico

En este capítulo se resumen los conceptos fundamentales en los que se basa el trabajo realizado. En la sección 2.1 se definen los indicadores de la función cardiaca en términos de la anatomía del corazón y se explica cómo cuantificarlos a partir de estudios cardiacos SAX MR-C. En la sección 2.2 se presentan las redes neuronales profundas, con particular énfasis en la arquitectura U-Net, considerada un estándar para el problema de identificación de estructuras en imágenes de resonancia magnética. Asimismo, se revisan algunos de los enfoques actuales que permiten incorporar información anatómica en los modelos basados redes profundas. Por último, en la sección 2.3 se introduce el *software* multiplataforma y de código abierto para visualización y procesamiento de imágenes médicas llamado *3DSlicer*, dentro del cual se desarrolló la herramienta propuesta en este trabajo.

2.1. Cuantificación de la función cardiaca

A lo largo del presente documento, se entenderá por cuantificación de la función cardiaca al proceso de obtención de diferentes indicadores o descriptores del funcionamiento y la estructura del corazón. Específicamente, este trabajo se centra en la cuantificación cardiaca a partir de la medición los volúmenes ventriculares, y el cálculo de la EF y LVM. La EF está definida en términos del volumen del ventrículo izquierdo (LVEF) y derecho (RVEF) del corazón, mientras que la LVM se define a partir del volumen del tejido del miocardio correspondiente al LV.

Anatomía y fisiología del corazón

Para el estudio y visualización de los diferentes órganos del cuerpo humano, es común utilizar los planos anatómicos coronal, sagital y horizontal (ver la Figura 2.1a). Esto permite observar cortes de interés de diferentes órganos y tejidos. Sin embargo, para las vistas del corazón es usual la utilización de otros planos rotados respecto a los planos anatómicos. Tres planos típicos que serán considerados en todo este trabajo son el plano apical, plano del eje corto y plano del eje largo (Figura 2.1b).

En la Figura 2.1c se muestra una imagen ilustrativa de la anatomía del corazón. El mismo está formado principalmente por cuatro cavidades. Las dos superiores son las

aurículas y las dos inferiores los ventrículos. Las aurículas reciben sangre desde las venas (cava superior y pulmonares) y se encargan de dirigirla mediante su contracción a las cavidades inferiores, el ventrículo izquierdo y el ventrículo derecho. Los ventrículos son los encargados de bombear esta sangre a los pulmones o al resto del cuerpo. El músculo que permite los movimientos de contracción y relajación necesarios para bombear la sangre es el miocardio, el cual se encuentra rodeado externamente por un tejido llamado pericardio.

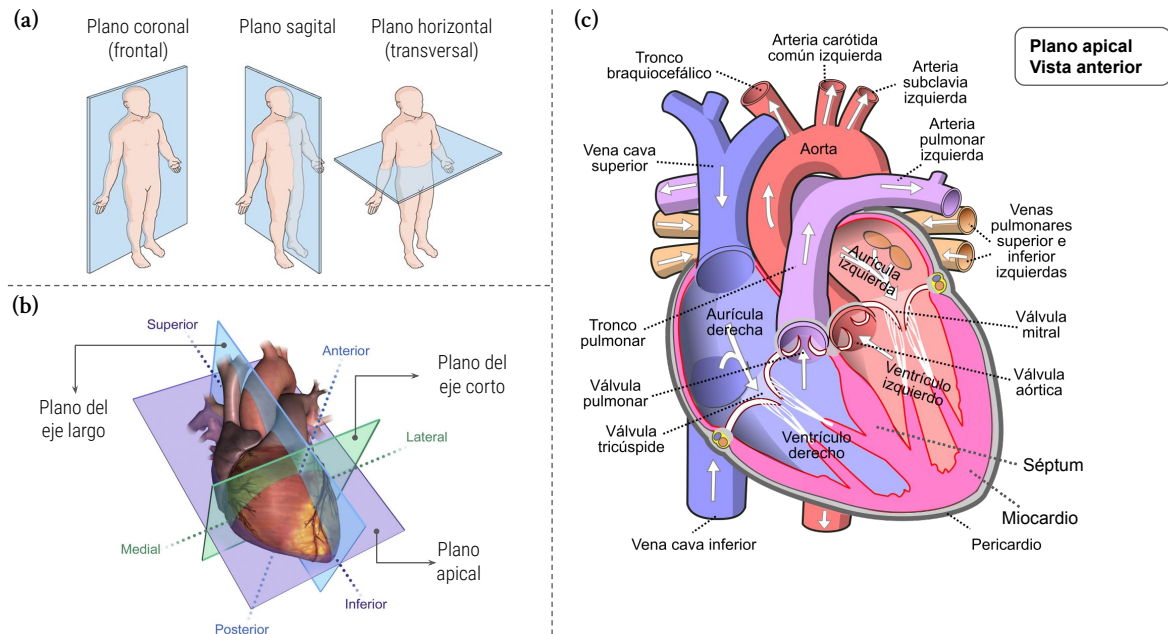


Figura 2.1: (a) Planos anatómicos del cuerpo humano^[14]. (b) Planos utilizados para visualizar las vistas 2D típicas del corazón^[15]. Estos planos están ubicados según un sistema de referencia fijo al corazón. (c) Anatomía del corazón humano^[16].

Durante el funcionamiento normal del corazón, el RV bombea sangre desoxigenada a través del tronco pulmonar, la cual se dirige a los pulmones a lo largo de las arterias pulmonares derecha e izquierda. Este tramo del circuito se denomina circuito pulmonar. Aquí ocurre un intercambio gaseoso dentro de los pulmones en el que la sangre libera dióxido de carbono en exceso y recibe oxígeno. Esta sangre oxigenada ingresa posteriormente a la aurícula izquierda a través de las venas pulmonares superiores e inferiores y es recirculada al LV. Una vez que ingresa al LV es bombeada al resto del cuerpo a través de la arteria aorta para llevar oxígeno a los diferentes órganos. A lo largo de este circuito, denominado circuito sistémico, la sangre va liberando oxígeno y recibiendo dióxido de carbono y otros desechos de las células de todo el cuerpo. Finalmente, el ciclo termina cuando la sangre desoxigenada ingresa a la aurícula derecha a través de las venas cava superior e inferior y es redirigida al RV para comenzar nuevamente.

Ciclo cardíaco

El ciclo cardíaco es el periodo de tiempo que incluye todos los eventos asociados con un latido del corazón, comenzando con la contracción de las aurículas y terminando con la relajación de los ventrículos. El periodo en el que el corazón se contrae mientras bombea la sangre se denomina sístole. El periodo de relajación en el que las cámaras del

corazón se llenan con sangre nuevamente se denomina diástole. Tanto los ventrículos como las aurículas tienen sus fases de diástole y sístole. En la Figura 2.2 se muestra un esquema del ciclo cardíaco de una persona sana, a una frecuencia de 75 latidos/min (lo que representa una duración aproximada de 0.8 s).

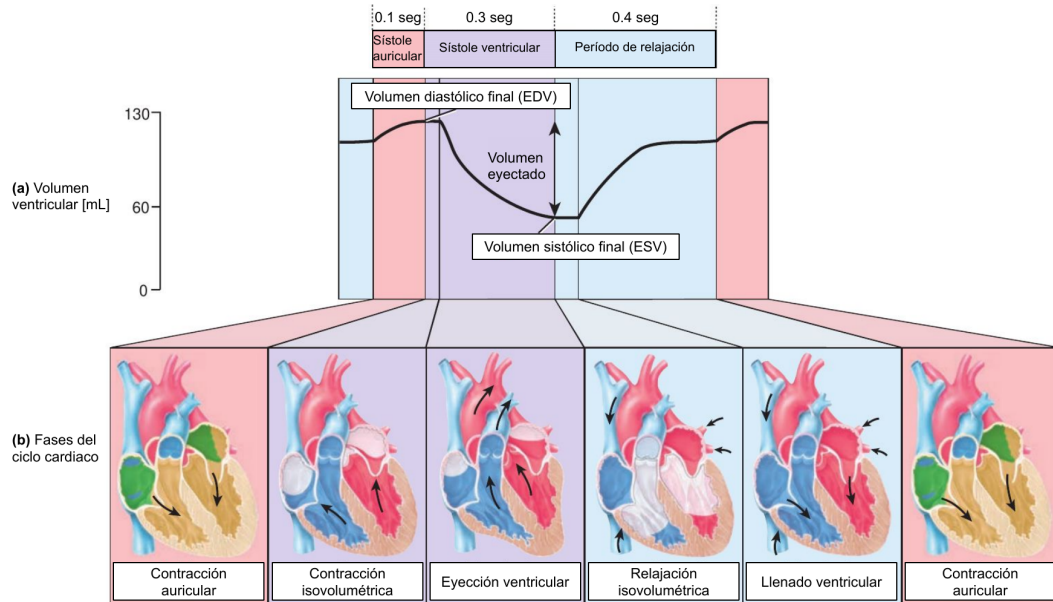


Figura 2.2: Esquema del ciclo cardíaco. (a) Variación de los volúmenes ventriculares. En este caso se muestra la curva del LV. (b) Fases del ciclo cardíaco desde la contracción auricular hasta la relajación y llenado ventricular. Imagen adaptada de^[17].

Como puede observarse en la Figura 2.2a, el volumen de los ventrículos varía de forma periódica entre dos valores extremos de la etapa de sístole ventricular, los cuales corresponden al volumen diastólico final y al volumen sistólico final.

Volumen diastólico final (EDV)

El ciclo cardíaco comienza en la etapa de sístole auricular, en la que las aurículas se contraen y la sangre se dirige hacia los ventrículos. Al mismo tiempo, los ventrículos están en fase de diástole y se van llenando de sangre. El volumen diastólico final corresponde al volumen de sangre en cada ventrículo al finalizar la etapa de sístole auricular, coincidente con la finalización de la fase de diástole ventricular (de ahí el nombre volumen *diastólico final*). Desde ese momento, las aurículas comenzarán su periodo de relajación y llenado mientras que los ventrículos comenzarán la etapa de contracción y eyección.

Volumen sistólico final (ESV)

La etapa de sístole ventricular comienza con una fase llamada contracción isovolumétrica. En este período de breve duración (~ 0.05 s), la presión de los ventrículos empuja la sangre contra las aurículas, forzando el cierre de las válvulas atrioventriculares (válvula mitral y tricúspide). Por otro lado, en ese período las válvulas semilunares se encuentran cerradas (válvula pulmonar y aórtica), quedando así las cuatro válvulas bloqueadas (ver la Figura 2.1c para mayor claridad). A partir de entonces, la presión en los ventrículos aumenta abruptamente con la contracción hasta superar valores críticos

que permiten la apertura de las válvulas semilunares. En este instante comienza la fase de eyección ventricular, en la que los ventrículos bombean la sangre hacia los pulmones y al resto del cuerpo a través de las válvulas recién abiertas. El volumen de sangre remanente en cada ventrículo luego de la eyección es el volumen sistólico final. A la diferencia entre los volúmenes diastólico y sistólico finales se la conoce como volumen eyectado.

Cuantificación de la función cardiaca

Los indicadores de la función y estructura cardiaca de mayor interés para este trabajo son la EF y la LVM, respectivamente. El cálculo de la EF requiere de la estimación del volumen eyectado. Por otro lado, la LVM se calcula en el tiempo de fin de diástole a partir del volumen del miocardio correspondiente al ventrículo izquierdo, donde la totalidad del séptum se considera parte del LV^[18] (ver Figura 2.1c).

Para el cálculo de la EF, en primer lugar se define el volumen eyectado (SV) como:

$$SV = EDV - ESV \quad (2.1)$$

donde EDV y ESV se calculan para los ventrículos izquierdo y derecho, obteniéndose así dos valores: $SV^{(LV)}$ y $SV^{(RV)}$. De esta forma, se define la EF según:

$$EF = 100 \times \left(\frac{SV}{EDV} \right). \quad (2.2)$$

Este indicador adimensional representa el porcentaje de volumen de sangre eyectado en un ciclo cardiaco. De igual manera que para el volumen eyectado, se obtienen dos valores para cada ventrículo: LVEF y RVEF. Acorde al consenso europeo del año 2019^[19], los valores de referencia en adultos entre 20 y 80 años de LVEF son 57~77 % para personas de sexo femenino y masculino. A su vez, para RVEF son 51~71 % (sexo femenino) y 52~72 % (sexo masculino).

Por otro lado, la masa del miocardio correspondiente al LV se calcula a partir del volumen del miocardio en el instante de telediástole ($EDV_{myo}^{(LV)}$) y la densidad del tejido miocárdico (ρ_{myo}), de la siguiente forma:

$$LVM = \rho_{myo} EDV_{myo}^{(LV)}, \quad (2.3)$$

donde el valor de ρ_{myo} aceptado es de 1.055 g/mL^[20]. Los valores de referencia de LVM^[19] varían entre 56~140 [g] (sexo femenino) y 92~176 [g] (sexo masculino).

Debido a que el tamaño y la masa del corazón varía entre diferentes personas, los volúmenes ventriculares y la masa de los mismos se suelen normalizar para poder comparar entre pacientes diferentes. Para ello se utiliza el coeficiente de superficie corporal (BSA), definido como:

$$BSA = \frac{\sqrt{M \cdot H}}{60} [m^2], \quad (2.4)$$

donde M es la masa del paciente [kg] y H es la altura [cm]. De esta forma, la LVM se normaliza de la siguiente manera: $LVM_{BSA} = LVM/BSA$ y se procede de forma análoga para los volúmenes.

2.2. Técnicas basadas en aprendizaje automático

Esta sección resume los conceptos de redes neuronales artificiales más relevantes en los que se basan las técnicas de detección de estructuras cardiacas utilizadas en el presente trabajo. En primer lugar, se presenta una breve introducción a los conceptos de *Machine Learning* y *Deep Learning*.

Machine Learning y *Deep Learning*

Las técnicas de aprendizaje profundo, más popularmente conocidas como *Deep Learning* (DL), son un subconjunto de técnicas basadas en redes neuronales artificiales que forman parte de una disciplina denominada *Aprendizaje Automático* o *Machine Learning* (ML). A su vez, ambas están incluídas un campo mucho más amplio conocido como *Inteligencia Artificial* (AI).

La definición de AI ha resultado ambigua desde que se pensara y soñara con computadoras programables capaces de poseer inteligencia propia. Si bien existen diversas definiciones, cada una brinda lugar al debate de conceptos como inteligencia, percepción, razonamiento, aprendizaje y conciencia, entre otros. El término fue acuñado por primera vez por el informático John McCarthy en 1956^[21], quien definió la AI como:

“La ciencia y la ingeniería de hacer máquinas inteligentes.”

Desde entonces, otras definiciones se fueron enunciando desde un punto de vista más práctico, orientado a la resolución y entendimiento de problemas reales. Una definición más actual es la siguiente^[22]:

“La capacidad de un sistema de interpretar correctamente los datos externos, de aprender de esos datos y de utilizar esos aprendizajes para lograr objetivos y tareas específicas mediante una adaptación flexible.”

En la Figura 2.3a se muestra un diagrama de *Venn* en el que se observa el orden de pertenencia de algunas de las disciplinas más populares incluídas en la AI.

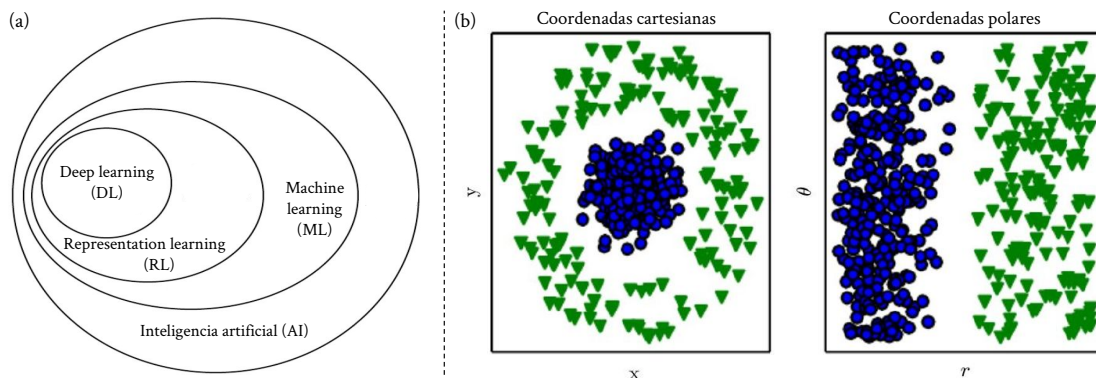


Figura 2.3: (a) Diagrama de *Venn* en el que se muestran diferentes disciplinas que forman parte del campo de la AI. (b) Dos representaciones diferentes de un mismo set de datos en un problema de clasificación binaria. Imágenes adaptada del libro *Deep Learning* de I. Goodfellow, Y. Bengio y A. Courville^[23].

En el campo de *Machine Learning* existen diversos algoritmos que permiten que un sistema *aprenda* en base a la experiencia. Más específicamente, en esta disciplina

se busca utilizar subconjuntos o muestras de datos para mejorar el desempeño de un sistema en una tarea dada. Estos sistemas deben ser capaces de generalizar su comportamiento mediante la inferencia a un conjunto más amplio de datos. El aprendizaje puede ser de forma *supervisada* (el sistema es provisto con datos y el resultado esperado de la tarea a resolver) o *no supervisada* (el sistema debe aprender por sí mismo a discernir entre resultados correctos o incorrectos). Para que esto sea posible, es necesario encontrar representaciones de los datos, como transformaciones lineales y/o no lineales, que permitan que el sistema extraiga características útiles para su aprendizaje. Por ejemplo, supóngase que se desea separar el conjunto de datos de la Figura 2.3b en dos categorías. Un algoritmo de ML debería ser capaz de asignar un triángulo verde o un círculo azul a cada punto según su clase. En este ejemplo se proponen dos representaciones: una en coordenadas cartesianas y otra en coordenadas polares. Si se elige sabiamente por la segunda, el problema se podría resolver aprendiendo una función lineal que divida al plano polar en dos. Si en cambio se elige la primera representación, el problema parecería ser más complejo y el modelo propuesto para resolver la misma tarea debería aprender una función no lineal, lo cual es más costoso.

Aprender la mejor representación de los datos puede transformar un problema complejo y difícil de resolver en uno más simple, como en el ejemplo de las coordenadas polares mencionado anteriormente. En el campo de *Representation Learning* (RL), se aplican algoritmos de ML para que el sistema aprenda representaciones óptimas de los datos en lugar de tener que ser encontradas de forma manual. Las representaciones aprendidas comúnmente resultan en mejores desempeños finales del sistema que aquellas brindadas manualmente.

Una dificultad del enfoque de RL es la extracción de características cada vez más abstractas a partir de los datos iniciales, que pueden ser necesarias para resolver problemas complejos. En este sentido, el campo de las *redes neuronales artificiales* (RNA) resuelve esta dificultad al proponer arquitecturas basadas en la interacción de unidades simples denominadas *neuronas artificiales*. Estas unidades se ordenan en capas y procesan la información de las neuronas de capas anteriores. De esta forma, van extrayendo características cada vez más complejas y abstractas de los datos iniciales. Con esta interacción, cada una de las capas del sistema puede aprender nuevas representaciones o características de los datos a partir de la información identificada en las capas previas. Los conceptos equivalentes de *deep learning*, *aprendizaje profundo* o *redes profundas* hacen referencia a RNAs compuestas por un gran número de capas, las cuales son capaces de resolver tareas de mayor complejidad.

Redes neuronales profundas

De todas las técnicas y disciplinas dentro del campo del ML, este trabajo se centra únicamente en las redes profundas. Específicamente, se estudian las redes convolucionales (CNN), haciendo hincapié en la arquitectura U-Net y en los *autoencoders variacionales* (VAEs).

Redes convolucionales

Las CNNs, originalmente propuestas en el trabajo de *Lecun et al*^[24], son un tipo particular de RNA ampliamente utilizadas en problemas relacionados a visión artificial cuyos datos son imágenes; aunque también son empleadas actualmente en otras

tareas que involucran el procesamiento de secuencias temporales, textos, audios o señales de diferentes naturalezas. En general, cualquier dato que pueda representarse con topologías cuadriculadas, en las que quede definida una dimensión espacial, puede ser procesado por estas redes. Por ejemplo, una serie temporal puede pensarse como una cuadrícula 1D, una imagen como una cuadrícula 2D o un volumen como una cuadrícula 3D, etc.

Las CNNs se caracterizan por el hecho que las unidades procesan la información presente en un *campo receptivo* de las neuronas de la capa anterior. De esta manera, las capas aprenden patrones locales con correlaciones espaciales y/o temporales de sus entradas, donde el término *espacial* se refiere a la topología mencionada anteriormente.

Mediante la concatenación de este tipo de capas, es posible que la red aprenda características cada vez más abstractas, llegando a identificar patrones de alta complejidad. Por ejemplo, en la Figura 2.4a se puede ver una CNN esquemática que logra identificar la presencia de un gato en una imagen. Para ello, las capas extraen características cada vez más complejas a medida que aumenta la profundidad de la red hasta aprender patrones complicados como un ojo, una nariz o una oreja.

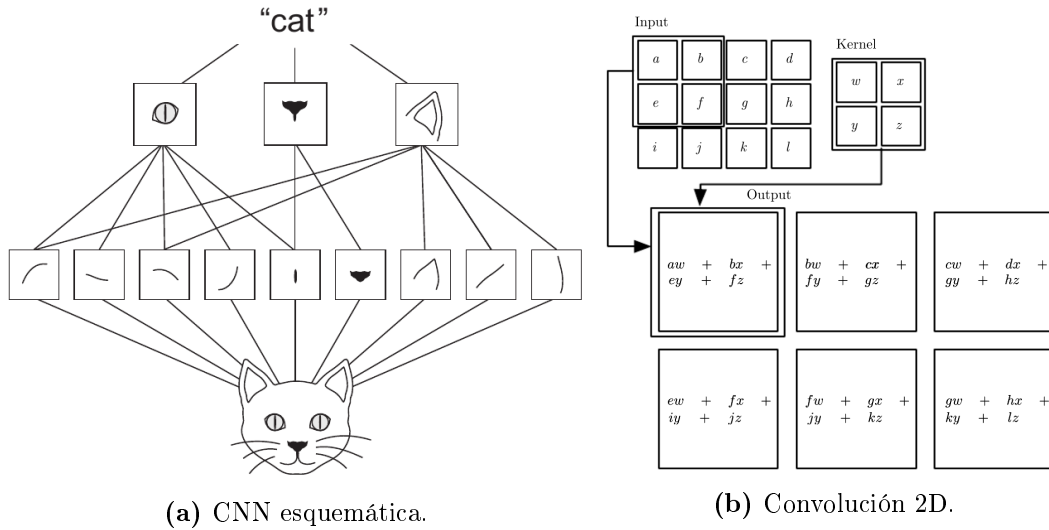


Figura 2.4: (a) Imagen esquemática de una CNN que identifica la presencia de un gato en una fotografía. Con la profundidad, las representaciones aprendidas se vuelven más complejas. (b) Operación de convolución 2D con un *kernel* (2,2). La salida se restringe a las posiciones en las que el *kernel* entra completamente en la imagen de entrada. Figuras extraídas de^[23].

El nombre de las CNNs hace referencia a la operación matemática de convolución que realizan sus unidades sobre sus entradas. Por ejemplo, en el caso de imágenes, la operación queda definida por:

$$S(i, j) = (I * K)(i, j) = \sum_m^M \sum_n^N I(i + m, j + n) K(m, n),$$

donde I es la imagen de entrada, K es el *kernel* de la convolución con tamaño (M, N) , mientras que i, j son los píxeles considerados. Esto puede verse gráficamente en la Figura 2.4b. Durante el entrenamiento, las redes convolucionales aprenden los *kernels* óptimos que extraigan las características más relevantes para resolver el problema particular en estudio, modificando gradualmente los componentes que conforman el *kernel*.

Además de la dimensión espacial que presenten los datos, éstos pueden contener diversos canales de información. Un ejemplo sencillo es el caso de las imágenes RGB, las cuales tienen dos dimensiones espaciales y tres canales de información para representar los colores. En el caso de datos con más de un canal, las capas convolucionales aplican un *filtro* que opera sobre todos ellos (formado por tantos *kernels* como canales haya). De esta forma, se genera una salida con tantos canales como filtros hayan sido aplicados. En la Figura 2.5 puede verse un ejemplo de forma gráfica, para un caso en el que se aplica una capa de convolución 2D con único filtro (3,3) a una imagen de entrada con más de un canal. A las salidas de las capas de convolución se las conoce comúnmente como *mapas de activación*.

Una práctica típica en las CNNs es ir agregando cada vez más filtros a medida que aumenta la profundidad de la red, con el fin de extraer cada vez más características para inferir objetos más complejos. Sin embargo, esto puede aumentar considerablemente el número de parámetros a entrenar y ser computacionalmente muy costoso, por lo que es común reducir la dimensionalidad espacial de los mapas de activación, por ejemplo, utilizando operaciones de *pooling* o cambiando el valor del *stride* de las capas convolucionales (el número de píxeles a desplazar el *kernel* sobre la imagen de entrada). Entre las operaciones de *pooling* más comunes se encuentran las de *max pooling* o *average pooling*, en las que se genera una salida de menor dimensión espacial que contiene en cada punto el máximo o el promedio de diferentes regiones de la entrada, definidas por una ventana de tamaño fijo (ver Figura 2.5b).

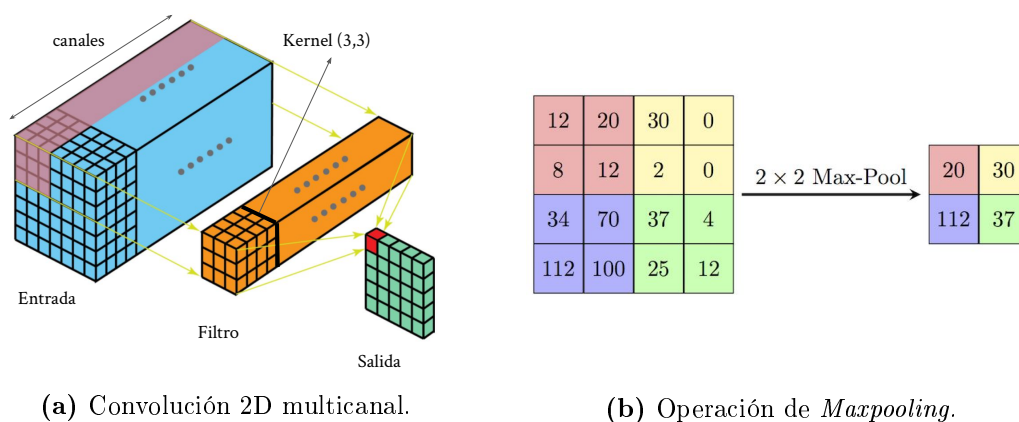


Figura 2.5: (a) Esquema de una convolución 2D con un filtro (3,3) aplicado sobre una entrada con muchos canales de información^[25]. (b) Operación de *Maxpooling* 2x2. Imagen de *Wikipedia: The Free Encyclopedia*, dominio público.

La aplicación de convoluciones descrita hasta el momento es una operación lineal. Para que las CNNs puedan aprender representaciones complejas es común la aplicación de funciones no lineales luego de la operación de convolución llamadas *funciones de activación*. Una de las más utilizadas debido a su buen desempeño en CNNs es la función *ReLU*, definida como $f(x) = x^+ := \max(0, x)$. Además de la mencionada, algunas funciones de activación típicas son la *tanh*, *sigmoide*, *softmax* y *LeakyReLU*, entre otras. En problemas de clasificación binaria, es común utilizar la función *sigmoide* para interpretar la activación de la neurona de salida como la probabilidad de pertenecer a una u otra clase, la cual es generalizada por la función *softmax* para el caso de problemas multi-clase.

Arquitectura U-Net

En muchas oportunidades, las CNNs son usadas en tareas de clasificación en las que las unidades de salida no están asociadas a una ubicación espacial de los datos de entrada. Sin embargo, en diversos problemas se desea clasificar la pertenencia de cada píxel a una clase dada, lo que se conoce como *segmentación semántica*. Por ejemplo, en el presente trabajo se desea asignar a cada píxel de las imágenes una etiqueta según éste sea parte del LV, RV o Myo.

La red U-Net^[11], cuya arquitectura se presenta en la Figura 2.6, ganó popularidad desde su aparición en el año 2015 gracias a su gran desempeño al resolver el problema de clasificación semántica en imágenes de estructuras neuronales de microscopía electrónica. A partir de entonces, se considera un estándar en las tareas de clasificación de estructuras en imágenes médicas.

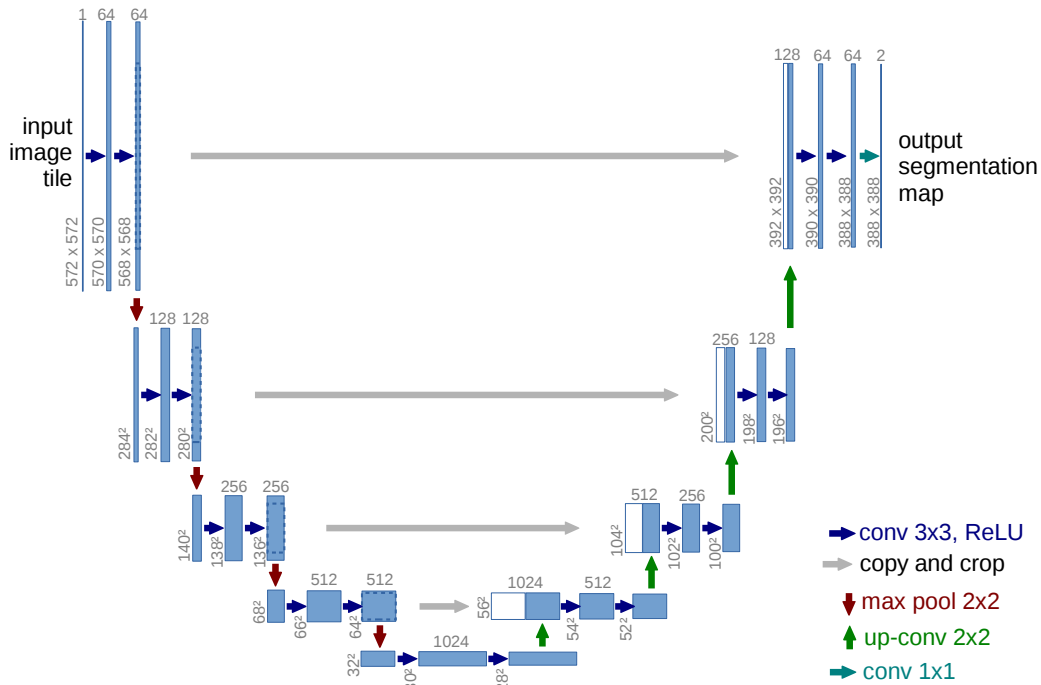


Figura 2.6: Arquitectura U-Net. Cada caja azul representa un mapa de activación multi-canal. El número de canales se muestra en la parte superior de cada caja, mientras que la dimensión (x,y) se muestra en la esquina inferior izquierda. Las cajas blancas representan mapas de activación copiados y las flechas denotan las diferentes operaciones. Imagen extraída del trabajo de *Ronneberger et al.*^[11].

Esta arquitectura en forma de U (de allí su nombre) se caracteriza por tener dos caminos de procesamiento de la información: uno de codificación y reducción de la dimensionalidad espacial y otro de decodificación y aumento de dimensión espacial. En el primero de ellos, sucesivas capas de convolución 2D son aplicadas (con filtros de tamaño $(3,3)$ y función de activación ReLu), seguidos de operaciones de *max pooling* 2×2 en las que se reduce la dimensión espacial a la mitad. A su vez, el número de filtros aplicados es cada vez mayor, lo que aumenta el número de canales de los mapas de activación de cada capa. En el camino de decodificación, se aplican operaciones de *upsampling* en las que se aumenta al doble la dimensión espacial en cada capa (*up-*

conv en la Figura 2.6), hasta obtener el tamaño inicial. Al mismo tiempo, se aplican operaciones de convolución 2D de forma simétrica al camino de codificación. De esta manera, la arquitectura ordena en distintos niveles a la información extraída de las imágenes originales, cada uno con una resolución espacial. Dado que solo se considera la información válida de las convoluciones realizadas, se debe realizar un *crop* o recortado, marcado con trazo discontinuo (Fig. 2.6), para las zonas en las que hubo pérdida de píxeles en los bordes luego de aplicar cada convolución.

Uno de los principales aportes del trabajo de *Ronneberger et al.* consiste en la combinación de la información de un nivel con la información procesada y decodificada del nivel inferior, lo cual se representa con flechas grises (Fig. 2.6). Esta operación permite que las capas de convolución posteriores del nivel aprendan a construir salidas más precisas. Adicionalmente, el hecho de contener una gran cantidad de canales en la etapa de *upsampling* facilita la propagación de información contextual a los niveles superiores.

Para realizar la clasificación final de cada píxel, en el trabajo original proponen el uso de la función *softmax*: $p_k(x) = \exp(\alpha_k(x)) / \left(\sum_{k'=1}^K \exp(\alpha_{k'}(x)) \right)$, donde $\alpha_k(x)$ representa la activación en el canal k para un píxel en la posición $x \in \Omega \subset \mathbb{Z}^2$, K es el número de clases y $p_k(x)$ es una aproximación de la probabilidad de que el píxel ubicado en x pertenezca a la clase k . Luego, se minimiza una función de pérdida (L) definida por la entropía cruzada:

$$L = \sum_{x \in \Omega} w(x) \log(p_{l(x)}(x)),$$

que penaliza la desviación de $p_{l(x)}$ de 1, siendo $l : \Omega \rightarrow \{1, 2, \dots, K\}$ la clase correcta de cada píxel. El término $w : \Omega \rightarrow \mathbb{R}$ es un mapa de pesos que introducen para darle a ciertos píxeles más importancia durante el entrenamiento de la red. En particular, para mejorar la precisión en la detección de los bordes de las estructuras neuronales que buscan identificar.

Otro aspecto importante de la arquitectura U-Net es el desempeño notable logrado con un set de datos considerablemente limitado, de tan solo 30 imágenes. En este sentido, se emplea la estrategia de *augmentación de datos* en la que se generan imágenes nuevas a partir de deformaciones sobre los datos de entrada (deformaciones elásticas, rotaciones, traslaciones, entre otras) que permiten a la red aprender propiedades de invarianza y robustez deseadas en el problema.

Autoencoders variacionales (VAEs)

Una arquitectura típica de las RNAs es el *autoencoder* (AE). El mismo está compuesto por un codificador $q_\phi : X \rightarrow Z$ y un decodificador $p_\theta : Z \rightarrow X$, siendo X el espacio de las variables de entrada y Z el *espacio latente*, donde los datos se encuentran codificados. Lo que se busca en estas redes es aprender a codificar y decodificar la información, logrando recuperar la entrada x a la salida. De esta forma, los parámetros del codificador y decodificador deben minimizar la diferencia entre la entrada (X) y la reconstrucción (X') de la siguiente forma:

$$\phi, \theta = \arg \min_{\phi, \theta} \|X - X'\|,$$

siendo $X' := (q \circ p)$.

A partir de este concepto se desarrollan los *autoencoders variacionales*, en los que se busca generar nuevos datos con características similares a los de entrada. Entrenando al codificador para que aprenda una distribución de probabilidad a posteriori $p(z|x)$, es posible tomar muestras \mathbf{z} del espacio latente y usar el decodificador para construir un modelo generativo que genere datos únicos en la salida, con propiedades similares a aquellos usados en el entrenamiento. En la Figura 2.7 se presenta esquemáticamente su arquitectura.

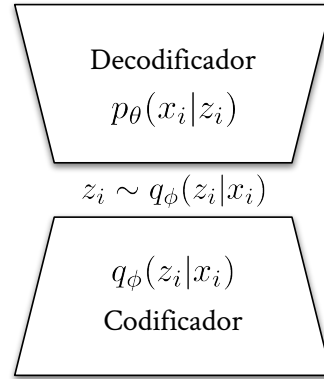


Figura 2.7: Arquitectura esquemática de un *autoencoder variacional* (VAE) compuesto por un codificador q_ϕ y un decodificador p_θ .

En modelos generativos, se desea optimizar la *likelihood* marginal $p_\theta(\mathbf{x}^{(i)})$ de los datos $\mathbf{x}^{(i)}$ muestreados. De acuerdo al trabajo de Kingma et al.^[26], dicha distribución queda definida según:

$$\log(p_\theta(\mathbf{x}^{(i)})) = D_{KL}(q_\phi(\mathbf{z}|\mathbf{x}^{(i)})||p_\theta(\mathbf{z}|\mathbf{x}^{(i)})) + \mathcal{L}(\theta, \phi; \mathbf{x}^{(i)}), \quad (2.5)$$

donde D_{KL} es la divergencia de Kullback-Leibler entre la distribución a posteriori aproximada $q_\phi(\mathbf{z}|\mathbf{x}^{(i)})$ y la teórica $p_\theta(\mathbf{z}|\mathbf{x}^{(i)})$, que resulta intratable en redes neuronales con no-linealidades. El segundo término \mathcal{L} es la denominada *cota inferior* o *variational lower bound*, que adquiere la expresión:

$$\mathcal{L}(\theta, \phi; \mathbf{x}^{(i)}) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}^{(i)})}[\log(p_\theta(\mathbf{x}^{(i)}|\mathbf{z}))] - D_{KL}(q_\phi(\mathbf{z}|\mathbf{x}^{(i)})||p_\theta(\mathbf{z}))), \quad (2.6)$$

cuyo primer término es el error de reconstrucción del decodificador y el segundo término es una medida de la información perdida al utilizar $q_\phi(\mathbf{z}|\mathbf{x}^{(i)})$ para representar $p_\theta(\mathbf{z})$. De esta forma, la optimización es posible maximizando la cota inferior en lugar de la log *likelihood* $\log(p_\theta(\mathbf{x}^{(i)}))$.

Bajo la hipótesis de distribuciones *gaussianas* a priori $p_\theta(\mathbf{z})$ y a posteriori aproximada $q_\phi(\mathbf{z}|\mathbf{x}^{(i)})$, se obtiene la siguiente expresión cerrada:

$$-D_{KL}(q_\phi(\mathbf{z}|\mathbf{x}^{(i)})||p_\theta(\mathbf{z}))) = \frac{1}{2} \sum_{j=1}^J (1 + \log(\sigma_j^2) - \mu_j^2 - \sigma_j^2), \quad (2.7)$$

siendo J la dimensión del espacio latente.

2.3. Software 3DSlicer

El entorno en el que se decidió implementar la aplicación de este trabajo es el *software* 3DSlicer. Esta herramienta de visualización y procesamiento de imágenes médicas de código abierto fue desarrollada inicialmente en 1999 en el laboratorio *Surgical Planning Lab* del MIT, EEUU. Posteriormente y hasta la actualidad, se continuó su desarrollo y mantenimiento con el apoyo del Instituto Nacional de Salud (NIH). Al día de la fecha, es ampliamente utilizada en numerosos trabajos de investigación, entre los que se incluye la incorporación de nuevas funcionalidades mediante el desarrollo de extensiones.

En la Figura 2.8 se muestra una imagen con la interfaz del *software* y sus diferentes elementos. Para la representación de los diferentes datos, *3DSlicer* utiliza el modelo MRML (*Medical Reality Modeling Language*). Este modelo implementa una librería compuesta principalmente por los paquetes VTK^[27] e ITK^[28], utilizados para la visualización, lectura/escritura y procesamiento de los datos. Además, utiliza la librería Qt para la interfaz de usuario (GUI).

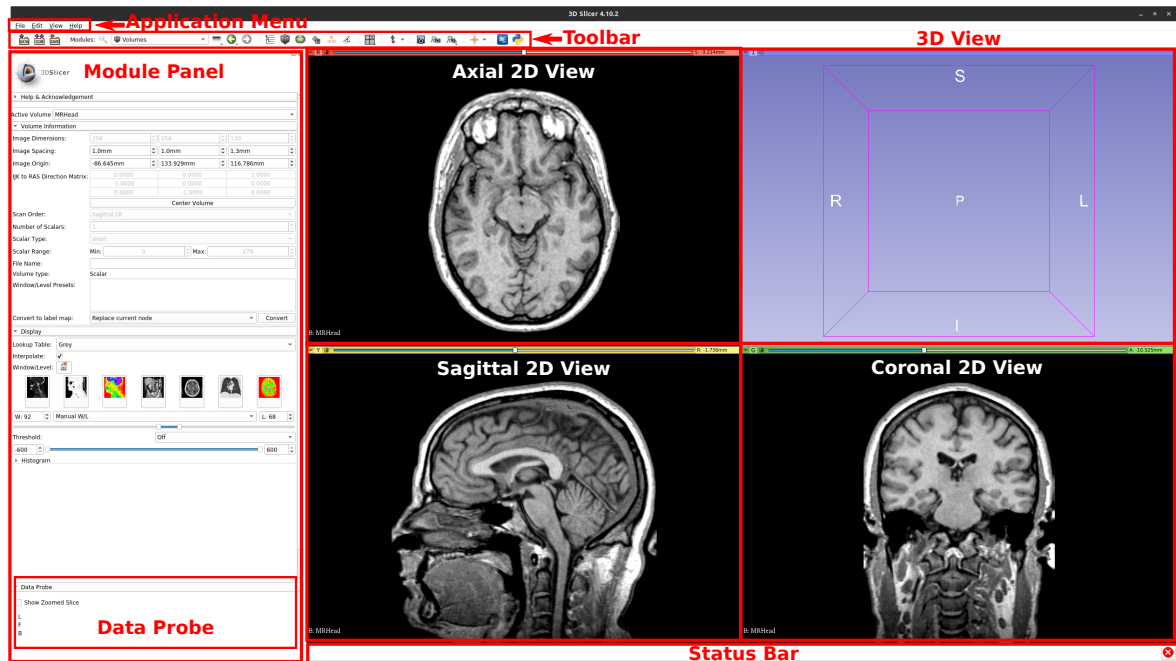


Figura 2.8: Interfaz de usuario del *software* 3DSlicer. Imagen extraída de la documentación oficial de *3DSlicer*.

Uno de los componentes más importantes del modelo MRML es la *escena* (*MRML scene*), la cual contiene una lista de nodos MRML. Cada nodo MRML posee un ID único que lo identifica en la escena, un nombre, atributos y otras propiedades adicionales para almacenar información según su tipo de datos. Entre los tipos de datos que almacenan los nodos se destacan los volúmenes de imágenes, set de puntos, transformaciones y segmentaciones. Es importante notar que todos los datos en *3DSlicer* tienen un nodo asociado, incluidos los de visualización (propiedades de la cámara 3D y controlador de vistas 2D). Una funcionalidad interesante de los nodos es su capacidad de invocar eventos cuando su contenido y su estado interno se modifican. Esto resulta útil para la implementación de *callbacks* que notifiquen a los diferentes módulos acerca del estado de la escena. Por ejemplo, si el usuario modifica un volumen desde la GUI, el nodo que

lo representa emite un evento de modificación que podría ser atendido y procesado por la lógica de los diferentes módulos.

Para la implementación de módulos y extensiones, *3DSlicer* sigue el patrón de diseño MVC (*Modelo Vista Controlador*), en el cual la representación interna de los datos es abstraída del usuario y la GUI (Figura 2.9). En este sentido, se implementan tres componentes interconectados entre sí: el Modelo (*Logic*), la Vista (*Widget*) y el Controlador (modelo MRML). En el componente *Logic* se gestionan los datos y la lógica de la aplicación, siendo ésta independiente de la interfaz de usuario. En *Widget* se implementa y maneja la información referida a interacciones del usuario y visualización de información. Por último, el modelo MRML se encarga de recibir entradas del usuario y vincularlas con los datos del componente *Logic*.

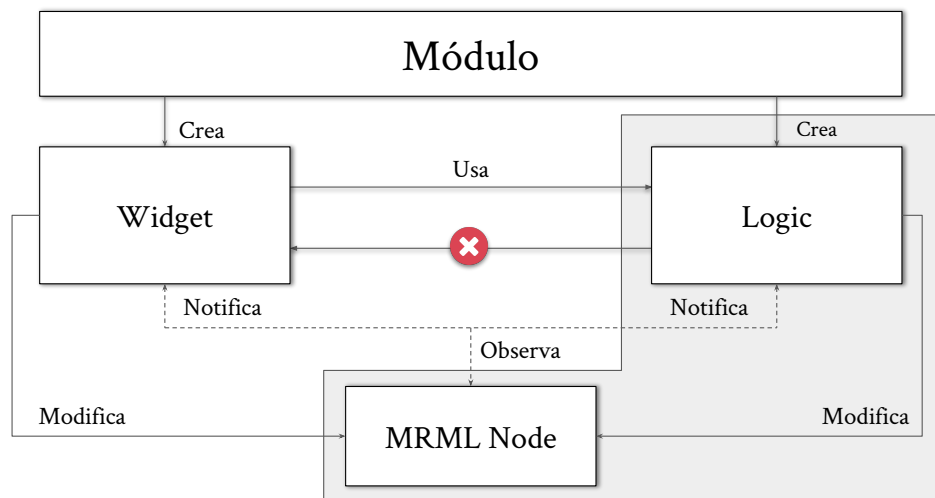


Figura 2.9: Patrón de diseño MVC utilizado en la implementación de módulos y extensiones de *3DSlicer*.

Capítulo 3

CardIAc: una aplicación para la cuantificación cardiaca

CardIAc es la herramienta *open source* desarrollada e integrada como una extensión al *software* 3DSlicer para cuantificar la función cardiaca a partir de la detección del tejido miocárdico y los ventrículos en estudios de resonancia magnética. Mediante la misma, es posible medir la EF, LVM, esfuerzo y deformación miocárdicos. La aplicación se compone de dos módulos, uno para la detección de estructuras cardiacas y cuantificación de la EF y LVM de forma automática, denominado AI Segmentation, y el otro para cuantificar el esfuerzo miocárdico, denominado Strain and Motion. En este trabajo se diseñó y desarrolló el primero de ellos, AI Segmentation. Este módulo provee la funcionalidad de realizar una segmentación semántica del LV, RV y miocardio correspondiente al LV de forma automática, como también la de cuantificar los indicadores de la función cardiaca considerados en este proyecto: LVV, RVV, EF y LVM. El segundo módulo fue desarrollado por el Lic. en Física Agustín Bernardo, en el marco de su proyecto de Maestría en Física Médica del Instituto Balseiro^[29]. Con el módulo Strain and Motion, se provee una solución automática para cuantificar el esfuerzo miocárdico a partir de estudios SAX MR-C, utilizando como *input* las segmentaciones de las estructuras cardiacas. De forma conjunta, se trabajó en la integración de ambos módulos y sus flujos de trabajo para obtener la aplicación funcional que se denominó CardIAc.

En la Figura 3.1 se muestra la interfaz gráfica del módulo propuesto, junto con un esquema que resume su funcionalidad. El mismo se encuentra dividido en tres secciones principales: una de creación secuencias, una de segmentación semántica y otra de cuantificación. La primera de ellas sirve para crear una secuencia temporal con los volúmenes que conforman el estudio SAX MR-C cargados previamente en 3DSlicer. La segunda sección es la responsable de procesar los volúmenes de la secuencia temporal utilizando el modelo propuesto de redes neuronales (ver cap 4) para identificar las estructuras de interés (LV, RV, y Myo). Por último, la tercera sección se utiliza para cuantificar los indicadores de la función cardiaca.

Considerando la premisa de una aplicación sencilla y fácil de usar, se resumió el flujo de trabajo típico en estas únicas tres secciones. Sin embargo, cada sección agrega opciones adicionales dependiendo la tarea a desarrollar. Por ejemplo, al realizar la

segmentación automática es posible que ocurran errores en la detección de los tejidos, con lo cual resulta fundamental la posibilidad de realizar una corrección manual de la misma. Esto puede ser llevado a cabo mediante la opción *Edit segmentation*, la cual habilita la utilización del módulo *Segment Editor* nativo de 3DSlicer para trabajar de forma manual sobre la detección. De la misma forma, es posible que el centro del LV no sea localizado correctamente por el módulo, algo fundamental para la detección de las estructuras de interés. Si fuera necesario corregir el centro del LV, la interfaz gráfica brinda esta funcionalidad mediante el botón *Select Center*. Para mayor detalle sobre la funcionalidad y uso de la extensión, se recomienda al lector revisar el manual de usuario disponible de forma online^[30].

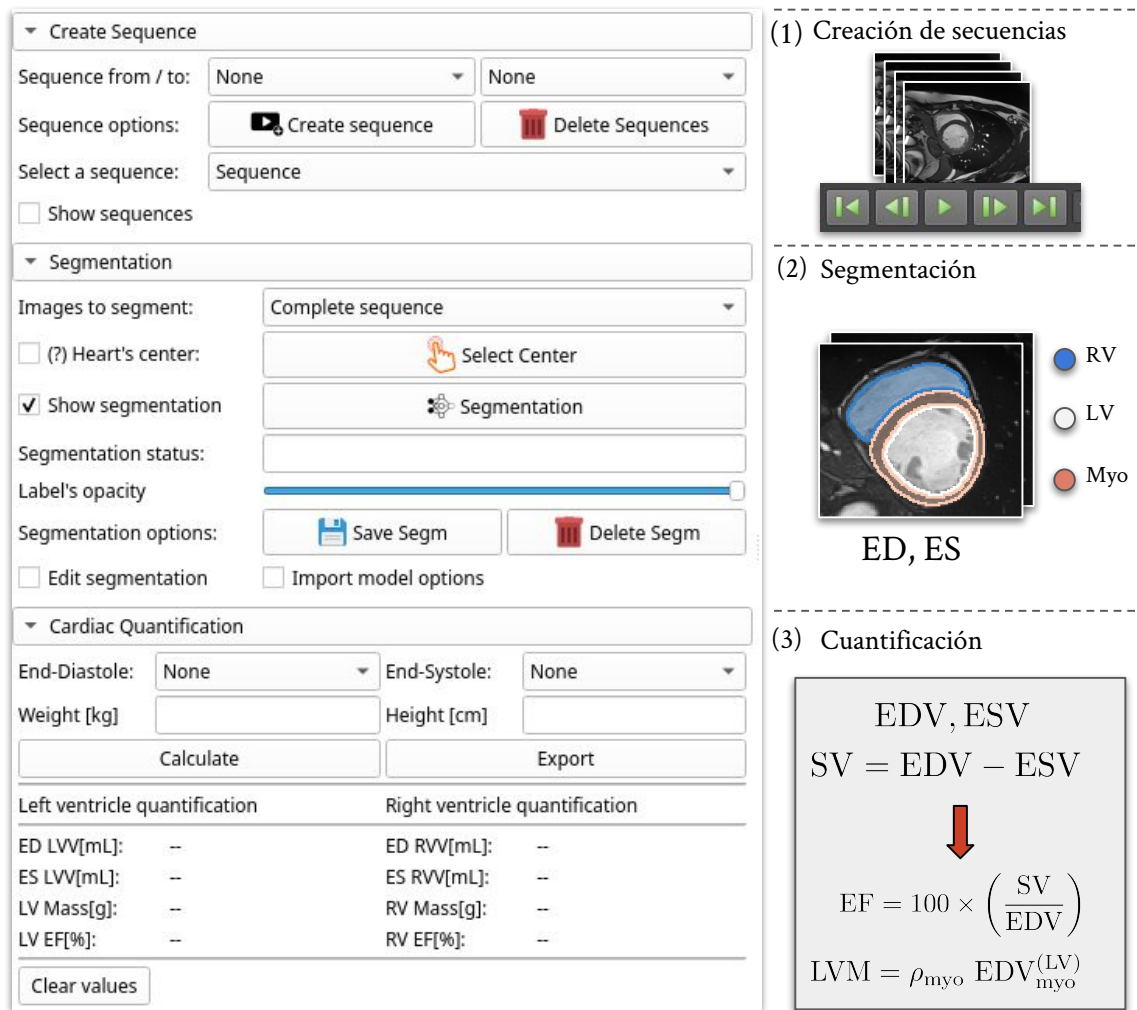


Figura 3.1: Presentación de la aplicación CardIAC: AI Segmentation. La herramienta está dividida en tres secciones principales: una de creación de secuencias temporales, otra de segmentación semántica y una última sección de cuantificación de los indicadores de la función cardiaca.

En la figura 3.2 se resume el flujo de trabajo normal al utilizar el módulo propuesto. En primer lugar (Fig. 3.2 (0)) se cargan los volúmenes del estudio SAX MR-C del paciente. Para ello, es posible “arrastrar” todos los archivos al programa o importarlos desde las opciones que provee 3DSlicer. Luego (Fig. 3.2 (1)), se crea una secuencia de volúmenes que conformarán el intervalo definido por los instantes inicial y final seleccionados. Para su creación, AI Segmentation utiliza la funcionalidad del módulo nativo

de 3DSlicer: Sequences, el cual permite crear, manipular y modificar los tipos de datos necesarios para el manejo de secuencias temporales. Para simplificar el uso del módulo, estas operaciones y configuraciones son abstraídas de la interfaz gráfica y realizadas de forma automática, de forma tal que al finalizar, es posible comenzar a reproducir la secuencia sin necesidad de ajustes adicionales. A continuación (Fig. 3.2 (2)), se realiza la detección de las estructuras de interés (segmentación semántica), para la cual debe indicarse si se desea segmentar el intervalo completo de la secuencia previamente creada, solo los instantes de telediástole y telesístole del ciclo, o únicamente un volumen del estudio. El proceso de segmentación es llevado a cabo por un sistema de dos redes neuronales profundas, cuya salida se corresponde con la identificación de las estructuras de interés (LV, RV y Myo). Una vez terminada la detección, las segmentaciones se sincronizan junto a las imágenes originales en una secuencia para habilitar al usuario una visualización conjunta en el tiempo. El último paso (Fig. 3.2 (3)) corresponde a la obtención de los indicadores de la función cardiaca. Aquí es necesario indicar los instantes de telediástole y telesístole solo en el caso que se haya segmentado la secuencia completa, ya que de otro modo quedan asignados automáticamente. Si el usuario desea obtener los indicadores normalizados por el coeficiente de superficie corporal BSA, debe ingresar la masa y la altura del paciente en los campos correspondientes.

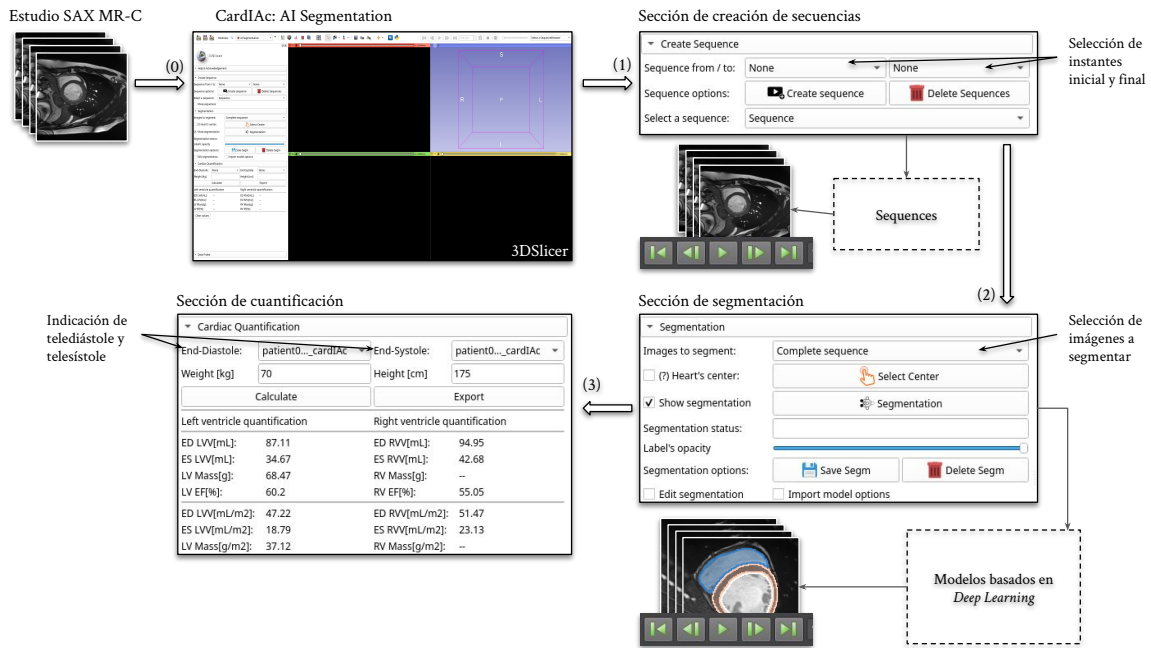


Figura 3.2: Flujo de trabajo normal de CardIAC: AI Segmentation. (0) Se ingresan las imágenes del estudio SAX MR-C de un paciente. (1) Se crea una secuencia temporal indicando el primer y último instante del ciclo cardiaco. (2) Se realiza la segmentación automática. (3) Se obtiene la cuantificación de los indicadores de la función cardiaca.

Desarrollo

En la etapa temprana de diseño y planificación de la aplicación se evaluó la idea de un desarrollo completo e integral desde cero. Finalmente, luego de un estudio de factibilidad, se decidió utilizar el *framework* provisto por el *software* 3DSlicer debido a su uso en la investigación, acceso al código fuente, disponibilidad de tutoriales, comunidad activa y principalmente, por la posibilidad de un desarrollo íntegro en lenguaje *python*,

uno de los más utilizados para *deep learning*. Sin embargo, se estudiaron diferentes alternativas a este *framework* como las aplicaciones *OsiriX*^[8], *Segment*^[9] y *cvi42*^[31], pero se descartaron debido a que en su mayoría son aplicaciones cerradas difíciles de extender.

Para el diseño del módulo propuesto se tuvieron en cuenta las recomendaciones de diseño para la implementación de extensiones que provee 3DSlicer. Para ello, desde los primeros prototipos se trabajó utilizando el patrón de diseño *Modelo Vista Controlador* que se presentó en la sección 2.3. En el esquema de la Figura 3.3 se resumen los diferentes paquetes desarrollados que componen a CardIAc: AI Segmentation. En el archivo principal se implementaron tres clases necesarias para incorporar el módulo como una extensión a 3DSlicer, las cuales deben heredar de clases definidas previamente en un *template* para desarrollo de módulos en 3DSlicer. La primera de estas clases sirve para definir datos como el título del módulo, desarrolladores, textos de ayuda y otra información general. La segunda y tercera clase se corresponde a los componentes *Widget* y *Logic*.

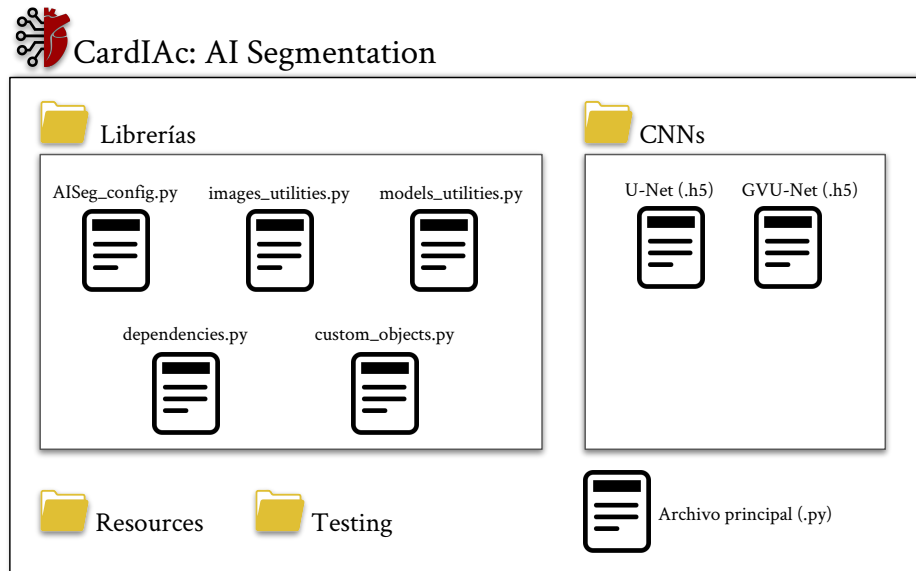


Figura 3.3: Paquetes y librerías que conforman la herramienta CardIAc: AI Segmentation.

Además del archivo principal, se creó un paquete con diversas librerías para encapsular la funcionalidad de la aplicación propuesta. Dentro del paquete se definió un archivo de configuración que contiene, entre otras cosas, la definición de las rutas de los diferentes modelos y parámetros necesarios para la inicialización del módulo. Las librerías `images_utilities.py` y `models_utilities.py` son las responsables de la funcionalidad básica utilizada para el procesamiento de imágenes y el manejo de los modelos de redes neuronales profundas. El manejo de las dependencias necesarias para instalar y utilizar el módulo propuesto se describe en el archivo `dependencias.py`, donde se definen las librerías externas a instalar automáticamente en caso de no estar instaladas, como *tensorflow* y *scikit-learn*. Por último, se creó una librería `custom_objects.py` en la que se describe la lógica para incorporar un modelo pre-entrenado de redes profundas alternativo al que se propone en este trabajo.

Los modelos entrenados se almacenan dentro de la carpeta CNNs como archivos con extensión “.h5”. El archivo U-Net.h5 contiene el modelo responsable de identificar el

centro del LV, mientras que el archivo GUV-Net.h5 contiene el modelo que se encarga de identificar las estructuras de interés (LV, RV, Myo) en las imágenes SAX MR-C. Finalmente, se cuenta con una carpeta *Resources* para incorporar recursos como íconos y otra carpeta *Testing* donde pueden desarrollarse programas para verificar el de funcionamiento de la aplicación.

Descripción detallada de la extensión propuesta para 3DSlicer

En la Figura 3.4 se muestra un esquema detallado en el que se representan las diferentes etapas del enfoque propuesto para cuantificar la función cardiaca.

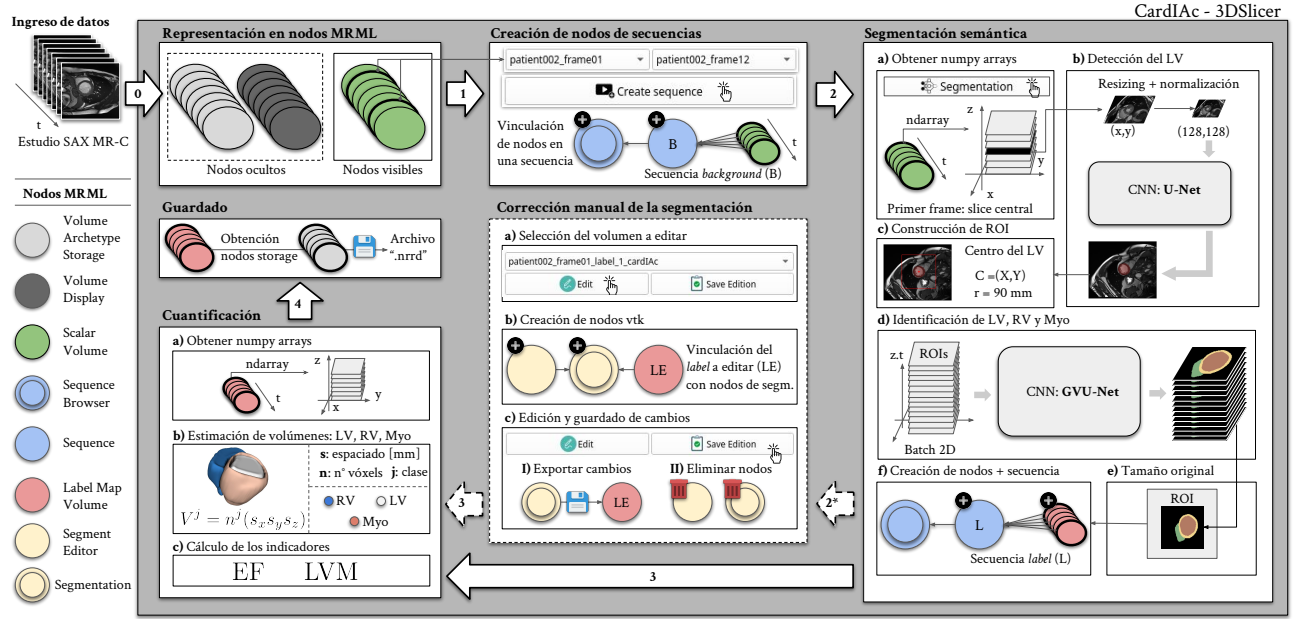


Figura 3.4: Enfoque automático propuesto para la cuantificación de la función cardiaca. La referencia de los Nodos MRML se muestra con colores en el margen izquierdo de la figura.

Una vez que el usuario ingresa el estudio SAX MR-C de un paciente a 3DSlicer (Fig. 3.4 paso 0), el modelo MRML se encarga de representar la información en nodos ocultos al usuario, con la información necesaria para el almacenamiento y visualización de los volúmenes; y nodos visibles, que contienen las imágenes del estudio. Para habilitar el uso de secuencias temporales (paso 1) se crea un nodo *Sequence*, al que se le vinculan los volúmenes contenidos en el intervalo definido por el usuario y un nodo *Sequence Browser*, que permite la reproducción temporal de las imágenes. De esta forma queda definida una secuencia formada por las imágenes del estudio, denominada *background* (indicada con la letra B en la figura).

A continuación se realiza una segmentación semántica del estudio contenido en la secuencia B y su resultado se guarda en una nueva secuencia denominada *label* (indicada con la letra L). Este paso se muestra en la Figura 3.4 (2). El proceso se divide en dos tareas: la construcción de una región de interés (ROI) que contenga únicamente al corazón (sub pasos 2a→2c) y la identificación del LV, RV y Myo (2d→2f). Inicialmente en (2a) se obtiene un arreglo de *numpy* correspondiente al plano central del primer volumen de la secuencia. Luego en 2b, se realiza un redimensionamiento de la resolución de la imagen a 128×128 píxeles y se aplica un pre-procesamiento para normalizar sus niveles de intensidad, restando la media y dividiendo por la desviación estándar

de intensidad de los píxeles. La imagen normalizada es la entrada de una CNN pre-entrenada que se encarga de identificar el LV (ver Cap. 4). A partir de la detección (**2c**), se calcula el centroide del LV y se construye una ROI rectangular de $180 \times 180 \text{ mm}^2$ cuyo centro es el centroide estimado, garantizando así que todo el corazón se encuentra dentro de la ROI. Una vez calculado el centro, se aplica la ROI para todos los volúmenes de la secuencia de la misma forma. Para la identificación de estructuras (**2d**), una segunda CNN (Cap. 4) procesa la información del conjunto de imágenes y entrega como salida la segmentación semántica. Luego de reacomodar los tamaños (**2e**), las clasificaciones obtenidas se guardan en nodos MRML para conformar la secuencia L (**2f**). Finalmente las secuencias B y L se sincronizan para poder ser reproducidas de forma conjunta en la interfaz.

En caso de ser necesario, es posible corregir manualmente las segmentaciones obtenidas (paso **2***). Inicialmente en (**2*a**), el usuario selecciona el volumen a editar (designado como LE en **2*b**) y dos nodos MRML son creados: uno que habilita y configura el módulo de edición y otro que contendrá la información de las modificaciones a realizar. Al finalizar la corrección, los cambios son escritos en el volumen LE y los nodos de edición son eliminados (**2*c**).

Para cuantificar los indicadores de interés (paso **3**), se obtienen los arreglos *numpy* contenidos en los nodos *Label Map Volume* y son utilizados para estimar el volumen de cada estructura $j \in \{LV, RV, Myo\}$ según $V^{(j)} = n^{(j)} s_x s_y s_z$, siendo n el número de vóxeles clasificados para cada estructura y s_x , s_y y s_z las resoluciones espaciales en [mm/pixel] en las direcciones x, y, z . A partir de los volúmenes, se cuantifica la EF y la LVM.

El enfoque propuesto contempla la posibilidad de guardar las segmentaciones obtenidas (**4**) como archivos con extensión “.nrrd”. Esto se realiza mediante los nodos de almacenamiento *Archetype Storage* asociados a los nodos *Label Map Volume*. Adicionalmente, se provee la facilidad de exportar la cuantificación de los indicadores LVV, RVV, EF y LVM como un archivo de texto.

Capítulo 4

Materiales y modelos de redes neuronales

4.1. Materiales

En este trabajo se utilizaron dos bases de datos (BD) para entrenar los modelos y evaluar su precisión. La primera es una base de datos pública (ACDC) y la segunda es una BD regional adquirida en el Sanatorio San Carlos (SSC) de San Carlos de Bariloche, Río Negro.

La base de datos pública está conformada por 100 pacientes provistos en la competencia *Automated Cardiac Diagnosis Challenge* (ACDC)^[32]. La misma está compuesta por un conjunto de 150 estudios SAX MR-C de pacientes con diferentes patologías (infarto de miocardio, miocardiopatía dilatada, miocardiopatía hipertrófica y ventrículo derecho anormal) y personas sanas. Además, ambos ventrículos y el tejido miocárdico del ventrículo izquierdo se encuentran delineados por un especialista experimentado en todos los instantes ED y ES. Las adquisiciones se obtuvieron durante un período de 6 años utilizando dos resonadores magnéticos, uno de 1.5 T de campo magnético (*Siemens Area*, *Siemens Medical Solutions*, Alemania) y el otro con 3.0 T (*Siemens Trio Tim*, *Siemens Medical Solutions*, Alemania). Las imágenes SAX MR-C fueron adquiridas en retención de respiración mediante la secuencia SSFP en orientación del eje corto con un grosor de 5 mm (a veces 8 mm). La resolución espacial se encuentra entre 1.37 y 1.68 mm²/píxel y el ciclo cardiaco está formado por 28-40 imágenes¹.

Por otro lado, la base de datos regional SSC está compuesta por 93 pacientes (52±17 años, de los cuales el 33 % son de sexo femenino) con diversas patologías (bloqueo de rama izquierda, infarto agudo de miocardio, miocardiopatía dilatada, miocardiopatía hipertrófica, miocardiopatía hipertensiva) y sanos. Los estudios fueron adquiridos durante 2 años con un equipo Philips Intera de 1.5 T en retención de respiración utilizando la secuencia SSFP. La resolución espacial promedio es de 1.25 mm²/píxel y la temporal de 20-30 imágenes. La delineación de las cavidades bajo estudio fue realizada por un especialista utilizando el *software Segment*^[9], con el cual además se cuantificó la EF, los volúmenes ventriculares en ED y ES y la masa del miocardio correspondiente al LV.

¹<https://acdc.creatis.insa-lyon.fr/description/databases.html>

4.2. Modelos de redes neuronales propuestos

El primer modelo CNN propuesto es una red U-Net compuesta por 4 niveles que se encarga de identificar el LV (Fig. 4.1). Se decidió utilizar esta arquitectura debido al buen rendimiento reportado en ^[12,13]. La entrada de esta red bidimensional es de 128×128 píxeles, donde se aplican cuatro bloques (uno por cada nivel) de dos convoluciones con un kernel de (3,3) seguidos de una función de activación ReLu y una capa *maxpooling* 2×2 (MP). La etapa de decodificación está formada por bloques similares en los que se aplican 2 capas de convolución (3,3) + ReLu, seguidas de operaciones de *upsampling* + convolución con activación lineal (UpConv). Finalmente, la salida es el resultado de una capa de convolución con un solo filtro seguido de una función de activación sigmoidal. De esta manera, los 64 canales son mapeados a un único canal que representa la *likelihood* de que un dado píxel pertenezca o no al LV. Solo los píxeles con *likelihood* > 0.5 son considerados como parte de dicha estructura.

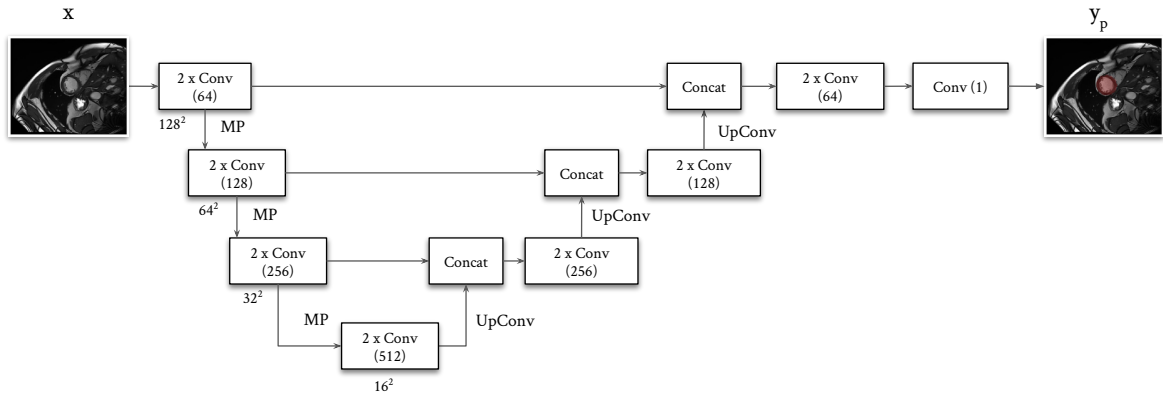


Figura 4.1: Red U-Net utilizada para la detección del LV. El número de canales se muestra en el interior de cada bloque, mientras que la dimensión espacial se muestra en su esquina inferior izquierda.

Para la detección del LV, RV y Myo, se propone la arquitectura que se muestra en la Figura 4.2 denominada Gvu-Net. Dicha arquitectura incorpora información anatómica del corazón para restringir el modelo U-Net mediante el uso de un *autoencoder variacional*. De esta manera, se pretende guiar el proceso de aprendizaje de la red U-Net hacia la obtención de estructuras con mayor plausibilidad anatómica.

El modelo generativo está formado por un codificador anatómico seguido por dos capas totalmente conectadas (FC) de 512 neuronas cada una. Mediante estas capas se codifica en el espacio latente la anatomía del corazón en la media μ y la log-varianza $\log(\sigma^2)$ de una distribución normal, cuya dimensión se fijó en 200. De este modo, el espacio latente queda parametrizado por las distribuciones N_x y N_y . Tomando muestras z_x y z_y de dichas distribuciones, es posible reconstruir nuevas segmentaciones semánticas usando el decodificador. Es importante notar que el mismo decodificador es utilizado tanto para z_x como para z_y .

A lo largo del entrenamiento, la reconstrucción obtenida con el modelo generativo (y_{z_x}) se combina con la predicción de la red U-Net (y_{p1}) mediante la concatenación de sus canales y la posterior aplicación de dos dos capas de convolución con 64 y 3 filtros. Finalmente, se aplica una función de activación sigmoidal para cada uno de los 3 canales, obteniendo así la *likelihood* de que un píxel pertenezca al LV, RV o Myo (y_{p2}).

Al igual que el primer modelo, solo los píxeles con *likelihood* > 0.5 son considerados como parte de cada estructura.

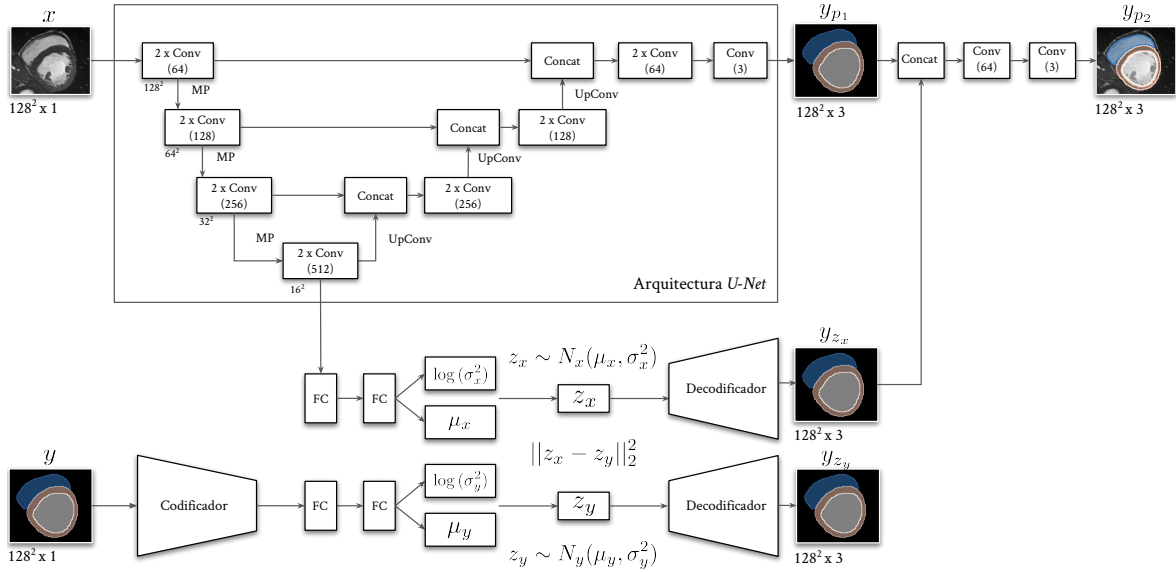


Figura 4.2: Red GUV-Net propuesta para la identificación del LV, RV y Myo. El número de canales se muestra en el interior de cada bloque, mientras que la dimensión espacial se muestra en su esquina inferior izquierda.

La información anatómica aprendida por el VAE puede ser utilizada para guiar el proceso de aprendizaje de la red U-Net, la cual puede centrarse en extraer características con mayor plausibilidad anatómica. Como se mencionó anteriormente, la incorporación del bloque inferior de la red GUV-Net correspondiente al modelo generativo se utiliza solo en la etapa de entrenamiento. Para la fase de predicción, se considera la entrada x y la salida de la red y_{p2} únicamente. Para lograr restringir el aprendizaje de la U-Net con características más plausibles, se definió la función de costo a minimizar como:

$$L = \frac{1}{N} \sum_i [\lambda_0 L_0(y^{(i)}, y_{p1}^{(i)}) + \lambda_1 L_1(y^{(i)}, y_{p2}^{(i)}) - \lambda_2 \log(p_\theta(y_{zx})) - \lambda_3 \log(p_\theta(y_{zy})) + \lambda_4 \|z_x - z_y\|_2^2], \quad (4.1)$$

donde λ_j define la contribución de cada pérdida L_j a la función global y $\log(p_\theta(y_{z_k}))$ es la log *likelihood* de cada VAE k . De forma similar a lo presentado en la sección 2.2, la optimización de dichas *likelihoods* es posible maximizando la cota inferior o *variational lower bound*:

$$\mathcal{L}_k(\theta, \phi; \mathbf{x}^{(i)}) = \mathbb{E}_{q_\phi(\mathbf{z}_k | \mathbf{x}^{(i)})} [\log(p_\theta(\mathbf{x}^{(i)} | \mathbf{z}_k))] - D_{KL}(q_\phi(\mathbf{z}_k | \mathbf{x}^{(i)}) || p_\theta(\mathbf{z}_k)), \quad (4.2)$$

donde la divergencia de Kullback-Leibler D_{KL} adquiere la expresión cerrada para el caso de distribuciones gaussianas (Ec. 2.7). Siguiendo la línea de los trabajos^[13,33], para los errores de reconstrucción \mathbb{E} de cada salida y_{z_k} y las pérdidas L_0, L_1 se propone el

uso de la distancia de *Jaccard* generalizada, definida como:

$$\text{GJD} = 1 - \frac{\sum_{j=1}^3 \min(y^{(j)}, y_p^{(j)})}{\sum_{j=1}^3 \max(y^{(j)}, y_p^{(j)})}, \quad (4.3)$$

siendo $y^{(j)}$ e $y_p^{(j)}$ las segmentaciones semánticas *ground-truth* y predichas para el LV, RV y Myo.

Entrenamiento

El primer modelo fue entrenado con la implementación *Adam* de *Keras* del gradiente estocástico descendiente, con una tasa de aprendizaje $lr = 3\text{e-}4$, mientras que para el modelo GUV-Net se usó el optimizador *Rmsprop* de *Keras* con $lr = 3\text{e-}4$. Las contribuciones de las pérdidas se definieron como $\lambda_j = 1$ para $j = 0, 1, 4$ y $\lambda_j = 1\text{e-}3$ para $j = 2, 3$.

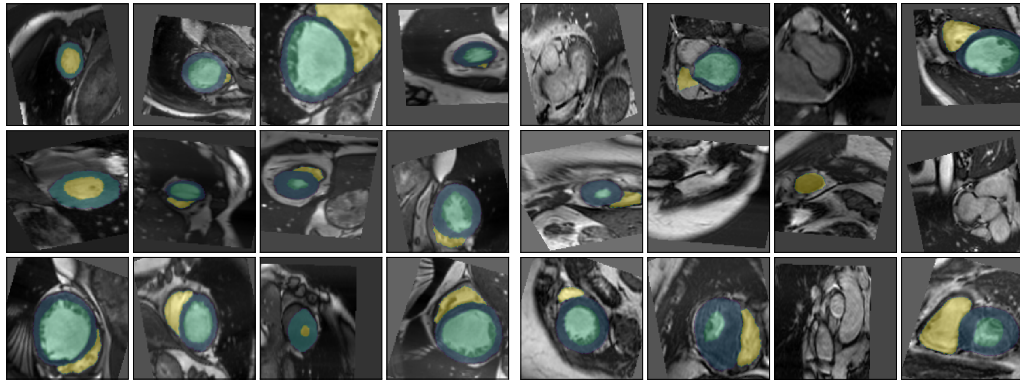
Para medir la precisión en la identificación de estructuras de interés se utilizó el coeficiente de *Dice*, definido como:

$$D(A, B) = 2 \frac{A \cap B}{A + B} \quad (4.4)$$

para dos áreas o volúmenes A y B . Este índice varía entre 0 (coincidencia nula) y 1 (coincidencia perfecta). Adicionalmente, se empleó la distancia de *Hausdorff* definida para dos áreas (HD 2D) o dos volúmenes (HD 3D) como:

$$HD(A, B) = \max(\max_{p \in A}(\min_{q \in B}(d(p, q))), \max_{q \in B}(\min_{p \in A}(d(p, q)))), \quad (4.5)$$

con d la distancia euclídeana. Una menor distancia de *Hausdorff* implica una mayor coincidencia. Es importante destacar que para mantener el modelo lo más simple posible se decidió no se introducir estrategias comúnmente utilizadas como *Residual learning*^[34], *Batch normalization*^[35] o *Dropouts*. Un modelo simple parece ser buena elección para ganar entendimiento acerca de la contribución de la información anatómica en la arquitectura.



(a) Base de datos: ACDC.

(b) Base de datos: SSC.

Figura 4.3: Ejemplos de imágenes generadas mediante la técnica de aumentación de datos.

Dado la escasa cantidad de datos presentes en las BD disponibles, para que la

red aprenda las características deseadas de invarianza y robustez se utilizó la técnica de aumentación de datos descrita en la sección 2.2. En este sentido, se aplicaron deformaciones sobre las imágenes de entrada que incluyen traslaciones del 20 % del tamaño de la imagen (altura y ancho), deformaciones de corte del 20 %, rotaciones del 20 % en el eje corto, un *zoom* de rango x3 e inversiones de la imagen (horizontal y vertical). En la Figura 4.3 pueden verse a modo de ejemplo algunas de las imágenes *aumentadas* generadas para ACDC y SSC, respectivamente.

Para normalizar los niveles de intensidad de las imágenes, se realiza un pre procesamiento antes del entrenamiento en el que se resta la media (μ) y se divide por la desviación estándar (σ) de la intensidad de los píxeles:

$$I \leftarrow \frac{(I - \mu)}{\sigma}, \quad (4.6)$$

donde el proceso se realiza independientemente para cada imagen (I) del set de datos.

Capítulo 5

Resultados

Se analizaron cuatro conjuntos de experimentos para validar la herramienta propuesta y cuantificar la función cardíaca. De esta manera, se agruparon los resultados en las siguientes secciones:

1. Construcción de la ROI: En esta sección se cuantifica el error en la localización del centroide del LV obtenida con el primer modelo U-Net en los datos ACDC y SSC.
2. Identificación de las estructuras cardíacas de interés: En este apartado se cuantifica el desempeño del modelo GVVU-Net entrenado con los datos ACDC y SSC. Se analizan las curvas de entrenamiento y se cuantifica el error en la detección de las estructuras de interés empleando el coeficiente de Dice y la distancia de *Hausdorff*. Por último, se realiza una comparación cualitativa sobre los datos ACDC entre la detección de las estructuras obtenidas con la red GVVU-Net (que incorpora información de *priors*) y con el modelo U-Net (sin esta información), el cual fue entrenado en este experimento para identificar el LV, RV y Myo.
3. Cuantificación de la función cardíaca: Aquí se analiza la precisión y los errores cometidos al cuantificar la función cardíaca de ambos ventrículos (volumen y EF) y la masa del LV.
4. Robustez de la cuantificación cardíaca: En esta última sección se estudia la robustez de la herramienta al cuantificar los indicadores de interés utilizando el modelo GVVU-Net que fue entrenado sobre los datos de ACDC y evaluado sobre la base de datos regional SSC.

Debido al reducido número de pacientes que conforman las bases de datos utilizadas, para entrenar los modelos se utilizó la estrategia de validación cruzada con 5 folds como se muestra en la Figura 5.1. De esta forma se logra emplear todos los estudios para medir el poder de generalización de la herramienta propuesta. Es importante remarcar que se optó por no ajustar hiperparámetros de los modelos, por lo cual no se utiliza un conjunto de validación.

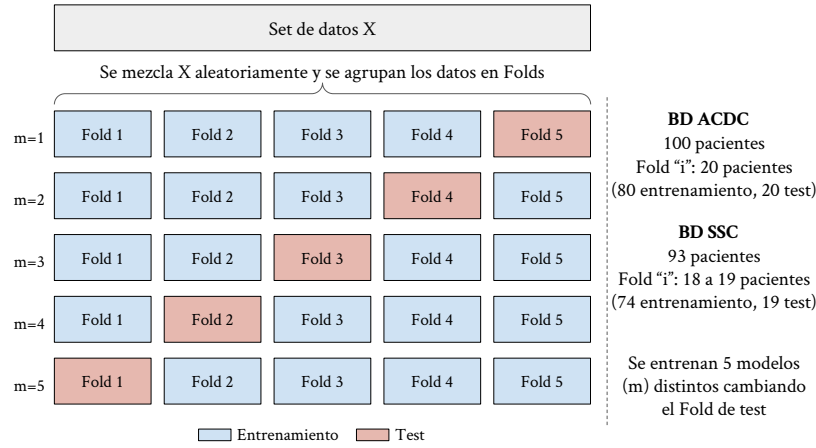


Figura 5.1: Esquema de validación cruzada con $k=5$ folds utilizada para entrenar los distintos modelos sobre las bases de datos ACDC y SSC.

5.1. Construcción de la ROI

La cuantificación del error en la construcción de la ROI se obtiene a partir de la norma euclídea entre el centroide del LV obtenido por la detección automática (c_p) y la delineación manual del mismo (c_{gt}). De esta forma, se obtuvo un error promedio de 1.34 píxeles (2.01 mm) para los datos de ACDC y de 2.41 píxeles (3.01 mm) para SSC.

Las Figuras 5.2a y 5.3a muestran que se comete un error sub-píxel para los datos ACDC (mediana 0.99 píxeles (1.51 mm)), aumentando a un error de 1.62 píxeles (2.03 mm) para SSC. Además, se puede observar (Figuras 5.2b y 5.3b) que la generación de la ROI es correcta incluso en el peor de los casos (19.93 píxeles para ACDC y 10.70 píxeles para SSC).

Es importante destacar que las principales variaciones entre los datos de ACDC y SSC se encuentran en la base y ápex del miocardio. Es por ello que la detección del centro del LV, que solo utiliza el plano central del corazón, se realizó con el modelo entrenado únicamente con los datos de ACDC. Como consecuencia de esto, se puede ver que el error cometido en la localización del LV para SSC es mayor. Sin embargo, este error resulta irrelevante ya que en todos los casos se logra contruir correctamente la ROI.

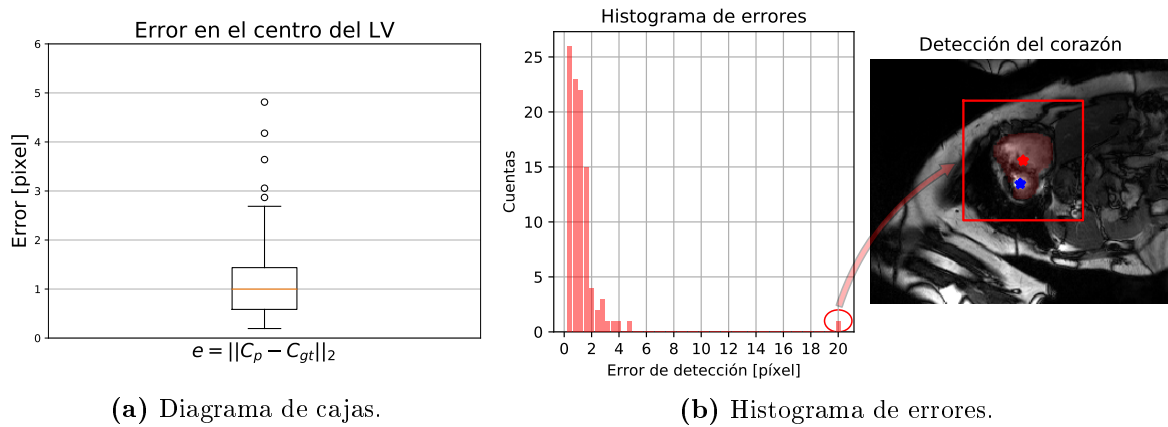


Figura 5.2: (a) Cuantificación del error en la detección del centro del LV (ACDC). (b) Histograma de errores y reconstrucción de la ROI para la detección con mayor error.

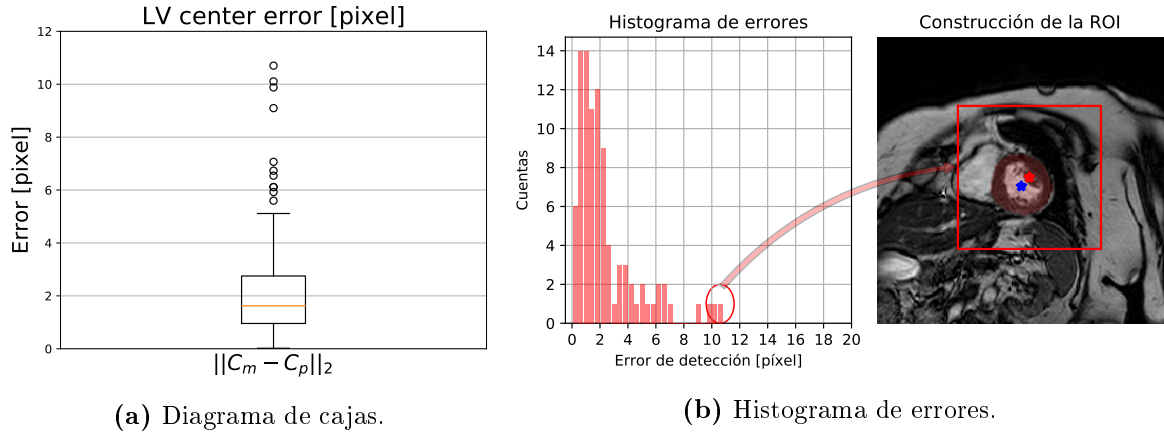


Figura 5.3: (a) Cuantificación del error en la detección del centro del LV (SSC). (b) Histograma de errores y reconstrucción de la ROI para la detección con mayor error.

Por último, en las figuras 5.4 y 5.5 se presentan algunos ejemplos de diferentes ROIs construidas. Como puede apreciarse, el método permite su construcción de forma correcta para ambas bases de datos.

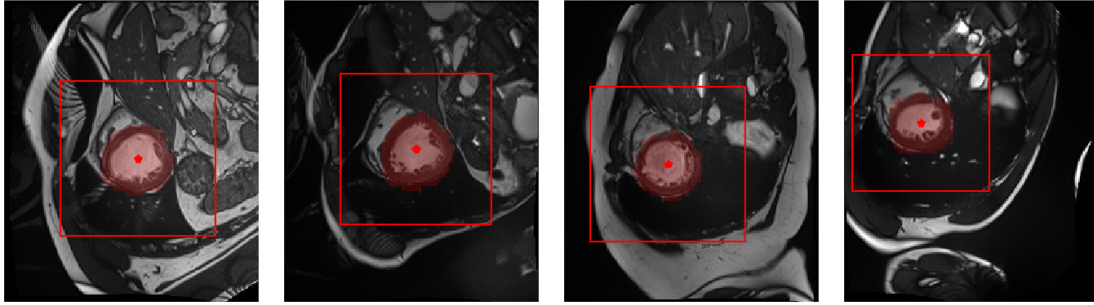


Figura 5.4: Ejemplos de regiones de interés (ROI) construidas para ACDC.

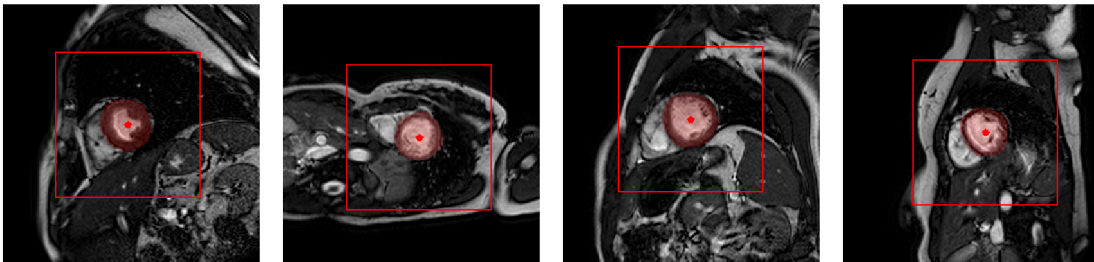


Figura 5.5: Ejemplos de regiones de interés (ROI) construidas para SSC.

5.2. Identificación de estructuras cardiacas

Una vez detectada la ROI donde se encuentra el corazón, se procedió a entrenar los modelos GUV-Net para detectar el LV, RV y Myo. Para reducir los efectos de aleatoriedad introducidos por el uso de GPUs, en el análisis de esta sección se entrenaron 3 modelos por cada fold. Es decir, se entrenaron un total de 30 modelos para los datos ACDC y SSC.

La Figura 5.6 muestra la evolución del aprendizaje del modelo propuesto para los datos de test de ACDC (izquierda) y SSC (derecha). La precisión y error promedio de los distintos modelos se representa con una línea negra, la cual esta dentro de los valores máximos y mínimos obtenidos de entre todos los modelos (recordar que hay 15 modelos para ACDC y 15 para SSC). La precisión final observada es mayor en los datos ACDC (Dice 0.909 ± 0.005) que SSC (Dice 0.893 ± 0.005), lo cual es consistente con un menor error obtenido para ACDC (Costo 0.62 ± 0.02) respecto a SSC (Costo 0.86 ± 0.07). Algunos resultados típicos de la clasificación de las estructuras de interés pueden visualizarse en la figura 5.7 (LV: rojo, RV: verde y Myo: amarillo), donde se muestran los estudios 036 de ACDC y 23294 de SSC.

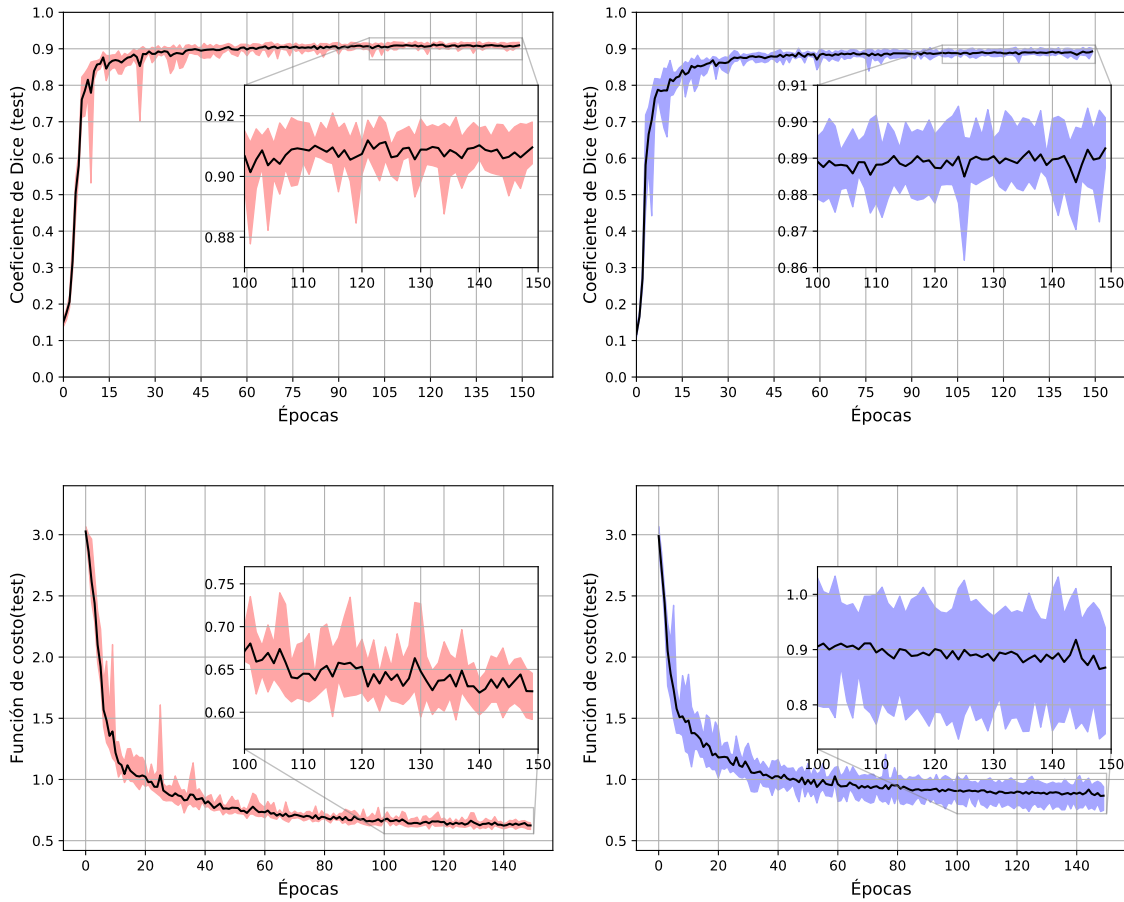


Figura 5.6: Curvas de aprendizaje para los datos de test del modelo propuesto GUV-Net entrenado sobre $k = 5$ folds (tres ejecuciones por cada fold). Se muestran la precisión y error para ACDC (izquierda) y SSC (derecha).

Para estudiar el efecto de la información anatómica o *priors* dentro del modelo propuesto, se entrenó una red U-Net igual a la utilizada dentro del modelo propuesto GUV-Net. Se utilizó el mismo esquema de *KFolding* con $k=5$ e iguales hiperparámetros, con lo que se obtuvo un valor de Dice de 0.901. En la Figura 5.8 se presenta el resultado de los estudios 028 y 054 para los modelos GUV-Net y U-Net, ambos de ACDC. Cualitativamente, puede notarse que la incorporación de información anatómica parece ayudar al modelo GUV-Net en la generación de estructuras anatómicamente plausibles, ya que algunas de las segmentaciones espurias (flechas blancas) se reducen (Fig.

5.8a). Sin embargo, si bien el modelo propuesto reduce las clasificaciones no deseadas, no logra eliminarlas completamente (Fig. 5.8b).

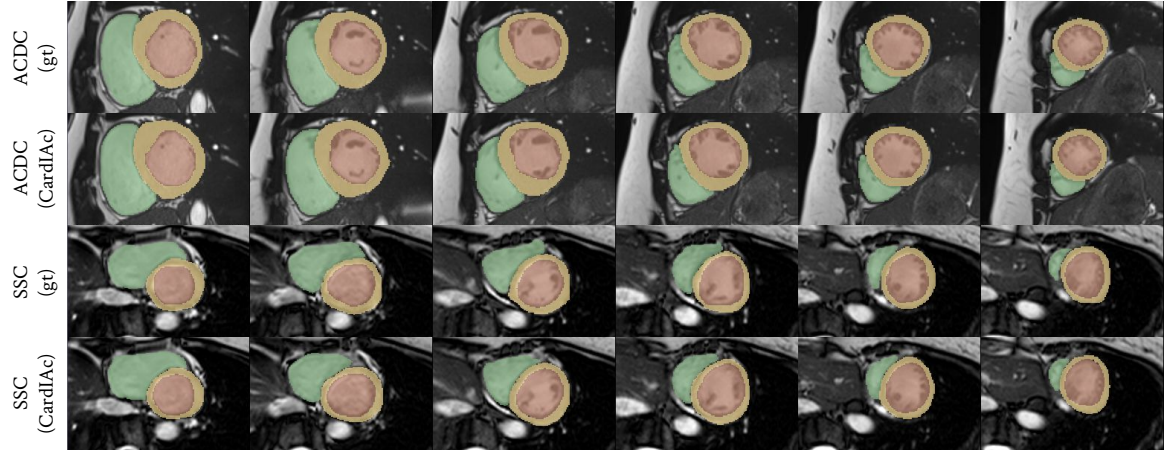


Figura 5.7: Resultados de la detección automática de las estructuras de interés para los estudios 036 de ACDC y 23294 de SSC, ambos en ED. Se muestran por columnas diferentes planos (eje corto) y por filas las segmentaciones *ground-truth* y predichas para cada caso.

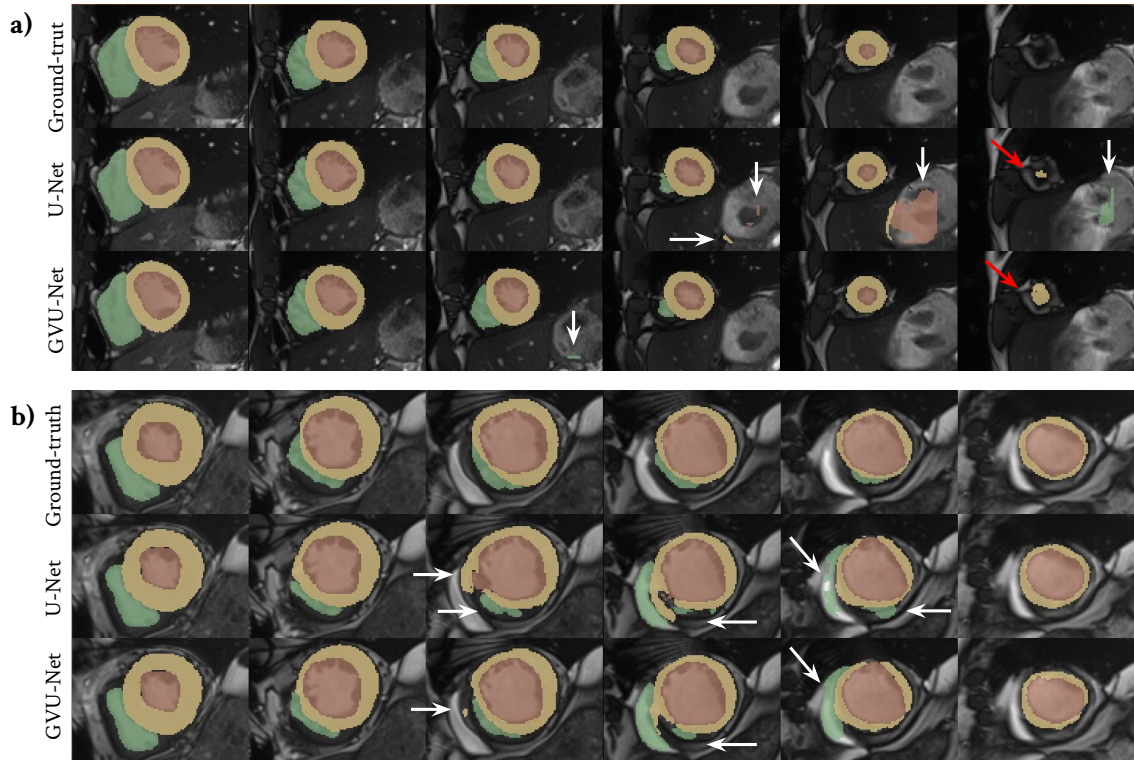


Figura 5.8: Segmentaciones obtenidas con la red U-Net (sin información previa) y con GUV-Net (con información previa) para los estudios 028 en ED (a) y 054 en ES (b) de los datos ACDC. Las flechas blancas indican las clasificaciones espurias mientras que las rojas señalan la detección del miocardio en la zona del ápex, incluso cuando esta no figura en la delineación manual.

Es interesante notar que ambos modelos CNN permiten identificar el tejido del miocardio correspondiente a la zona del ápex, aún cuando este no figura en la delineación manual de la BD pública (flechas rojas en 5.8a). Si bien el resultado que se obtiene es

más preciso que el *ground-truth*, esto afecta negativamente disminuyendo el valor de Dice. La BD regional, en cambio, sí contempla estas zonas del ápex. Asimismo, también incluye parte del miocardio y del RV en la zona basal, donde la morfología de ambos ventrículos cambia notoriamente (Figura 5.9). Identificar las estructuras en estos casos añade una dificultad adicional al problema, lo cual explica en parte que la precisión observada con los datos SSC sea menor que con ACDC.

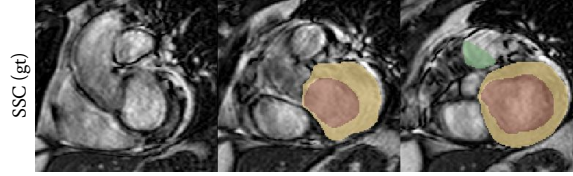


Figura 5.9: Segmentaciones *ground-truth* en la zona basal del estudio 17313 (ED) de SSC. La BD SSC contempla las delineaciones de parte del miocardio (amarillo) y del RV (verde) en esta zona donde la morfología de los ventrículos cambia considerablemente.

Precisión volumétrica

Con el objetivo de estudiar la precisión y el error de la clasificación en todo el volumen del corazón, se calculó el coeficiente de Dice y la distancia de *Hausdorff* de forma volumétrica para los instantes de ED y ES, y se los promedió para cada paciente de ACDC y SSC (Tabla 5.1). Se puede observar que la precisión disminuye para los datos regionales (SSC), lo cual se debe principalmente a que se contemplan las delineaciones manuales de mayor complejidad en la zona basal y del ápex previamente mencionadas.

BD	Dice (std)	HD (std)
ACDC	0.905 (0.065)	12.22 (15.39)
SSC	0.890 (0.058)	14.53 (16.64)

Tabla 5.1: Precisión de la segmentación cardíaca para las bases de datos ACDC y SSC.

Para cuantificar el desempeño de los modelos de forma más específica, se calculó la precisión (Dice) y el error (HD) discriminando la estructura cardíaca y los instantes ED y ES (Tabla 5.2). A partir de los resultados se puede ver que la segmentación presenta su mejor desempeño en la identificación del LV en telediástole (ED). Del mismo modo, la precisión más baja se obtiene al identificar el RV en telesístole. Esto es consistente con lo observado en la bibliografía y con el hecho que en el LV (ED) el miocardio tiene una morfología más compacta y simple que en el RV, lo cual facilita su detección.

	LV				RV				Myo			
	ED		ES		ED		ES		ED		ES	
BD	Dice (std)	HD (std)	Dice (std)	HD (std)	Dice (std)	HD (std)	Dice (std)	HD (std)	Dice (std)	HD (std)	Dice (std)	HD (std)
ACDC	0.96 (0.03)	6.05 (10.31)	0.92 (0.07)	9.51 (13.65)	0.92 (0.05)	17.19 (19.47)	0.86 (0.08)	17.19 (18.33)	0.88 (0.05)	10.47 (11.47)	0.89 (0.04)	12.90 (13.57)
SSC	0.94 (0.03)	9.28 (11.85)	0.89 (0.07)	13.24 (16.41)	0.90 (0.05)	12.52 (17.38)	0.86 (0.06)	16.50 (21.03)	0.86 (0.04)	17.05 (13.56)	0.88 (0.04)	18.62 (16.21)

Tabla 5.2: Precisión de la segmentación cardíaca por tejido y por instante temporal (ED y ES) para las bases de datos ACDC y SSC.

La variabilidad de los índices Dice y HD se puede visualizar en los gráficos *violinplots* de las Figuras 5.10 y 5.11. Estos gráficos permiten observar la distribución del error

en todo el rango de valores, junto con la mediana y los valores extremos. Puede verse que la precisión del modelo tiende a disminuir en el instante de ES, tanto en ACDC como SSC, donde se observa mayor dispersión y extremos más alejados, como se hizo evidente en la sección anterior.

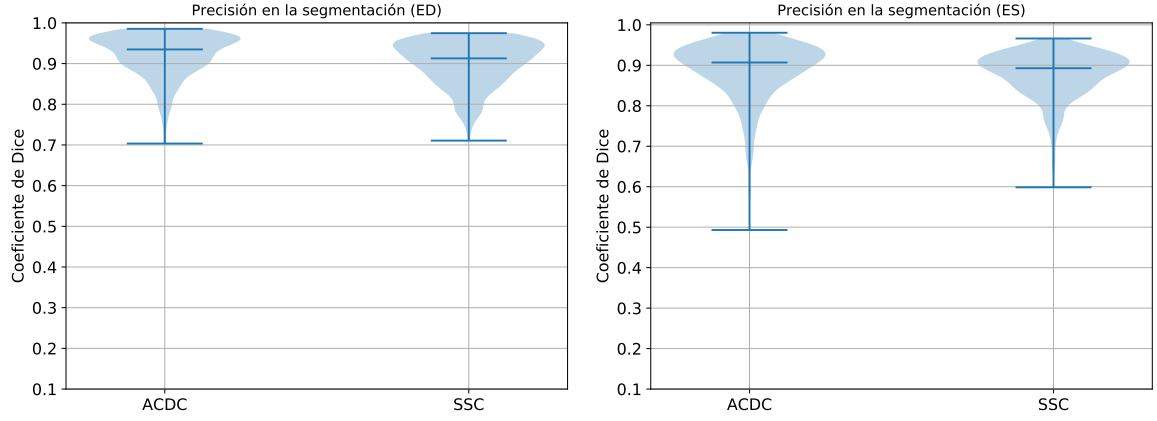


Figura 5.10: Variabilidad del coeficiente de Dice entre los datos de las BD ACDC y SSC para los modelos con arquitectura GVV-Net entrenados.

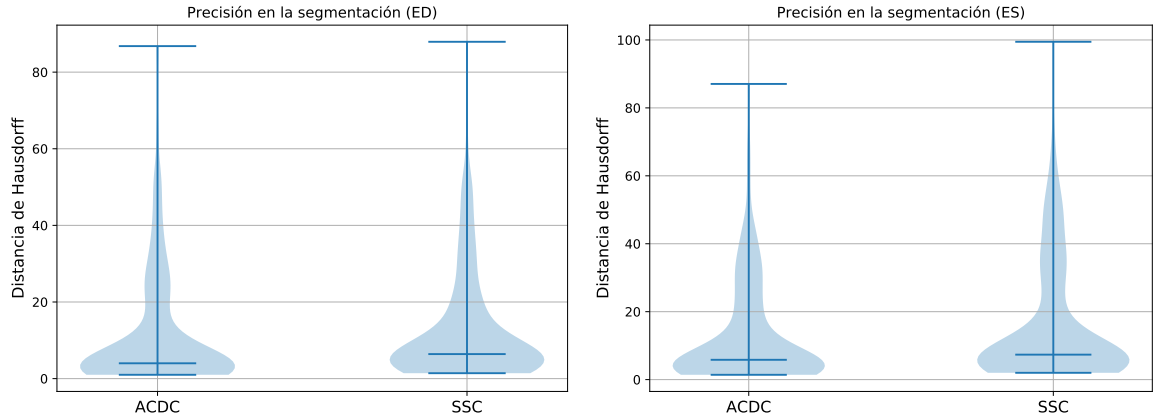


Figura 5.11: Variabilidad de la HD entre los datos de las BD ACDC y SSC para los modelos con arquitectura GVV-Net entrenados.

Finalmente, se destaca que el tiempo de segmentación obtenido con CardIAC para un estudio completo del ciclo cardíaco (típicamente formado por alrededor de 30 volúmenes, cada uno con 14 o 15 cortes o *slices*) es de: ~ 30 segundos en GPU y ~ 2 minutos utilizando solo la CPU. Esta mejora representa una reducción considerable en el procedimiento de cuantificación respecto de una cuantificación manual (~ 20 min por paciente solo para ED y ES^[10])

5.3. Cuantificación de la función cardíaca

Para observar la precisión en la cuantificación de la función cardíaca, se compararon los resultados obtenidos por un especialista contra los obtenidos de forma automática

con el método propuesto. En las Figuras 5.12 y 5.13 se presentan los resultados de dicha cuantificación y se estima el coeficiente de *Pearson* mediante un ajuste lineal para los datos ACDC y SSC. Se puede ver que el desempeño del método es satisfactorio para ambas bases de datos, ya que se obtiene una alta correlación respecto a la cuantificación realizada por un especialista en la LVM ($R_{ACDC} = 0.99$ y $R_{SSC} = 0.93$) y LVEF ($R_{ACDC} = 0.97$ y $R_{SSC} = 0.95$). Por otro lado, la disminución en el indicador RVEF ($R_{ACDC} = 0.88$ y $R_{SSC} = 0.82$) se debe a la menor precisión obtenida en la identificación del RV mencionada anteriormente.

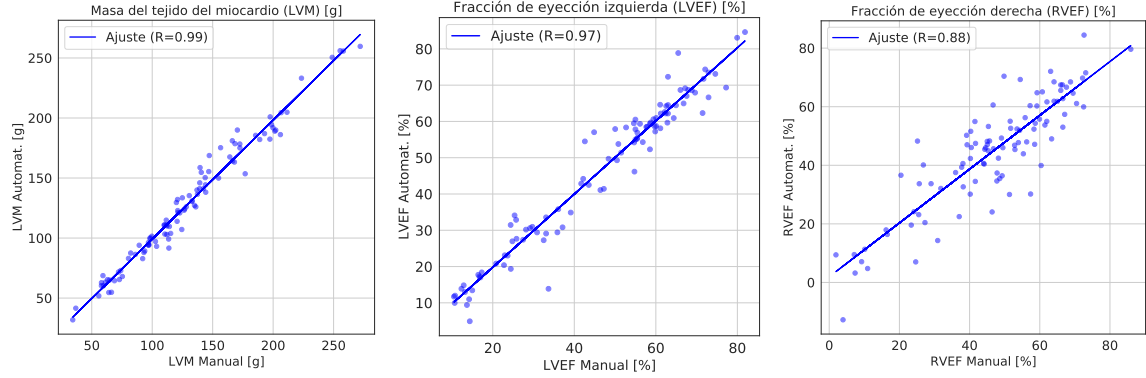


Figura 5.12: Comparación entre la cuantificación manual y automática con el método propuesto para los datos ACDC. Se muestra un ajuste lineal con el coeficiente de Pearson R .

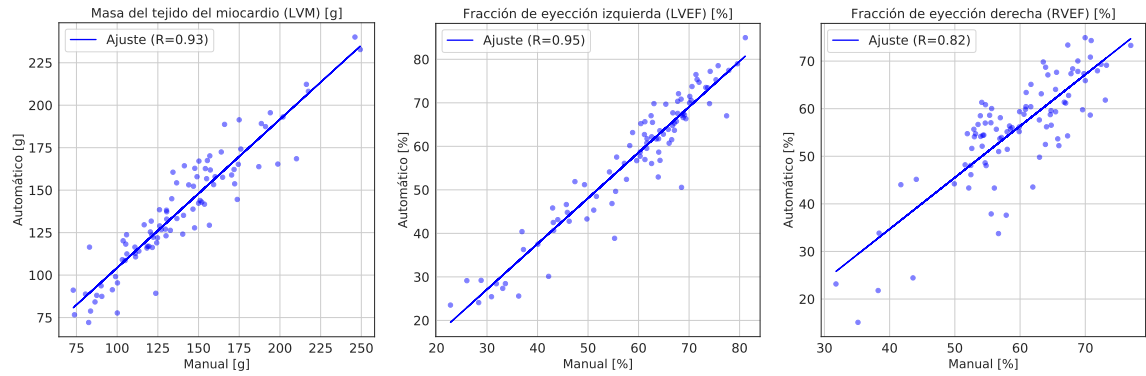


Figura 5.13: Comparación entre la cuantificación manual y automática con el método propuesto para los datos SSC. Se muestra un ajuste lineal con el coeficiente de Pearson R .

A continuación, se realiza un análisis estadístico para estudiar si existen diferencias significativas entre la medición automática y la manual. Para tal fin se utiliza un test de *Student* o t-test para dos muestras relacionadas (hipótesis nula de igual valor medio). En la Tabla 5.3 se resume la cuantificación obtenida por la extensión propuesta y la manual donde se incluye también el valor p -value del t-test y el coeficiente de *Pearson*. Asumiendo un valor de significancia $\alpha = 0.001$, es posible observar que no hay diferencias significativas entre las mediciones realizadas con las técnicas manual y automática (p -value $> \alpha$) para las diferentes variables a excepción del LVV en los datos públicos (ED y ES), y del RVV (ED) y RVEF en los datos regionales (p -value $\leq \alpha$). Si bien para estos casos es posible distinguir estadísticamente entre una u otra cuantificación

(Manual vs CardIac), esto no necesariamente implica que se tenga una mala precisión en la cuantificación, ya que las mediciones pueden ser estadísticamente diferentes pero la diferencia entre sus medias (o error) ser muy pequeña. En este sentido, se necesita hacer un análisis más profundo para observar el error entre ambas mediciones. Para ello se analizó el error al cuantificar dichos indicadores (Fig. 5.14 y 5.15) mediante gráficos de *Bland-Altman*, donde se muestra su valor medio y los intervalos de confianza ± 1.96 de la desviación estándar (std).

		Manual	CardIac	p	R
ACDC	LVV ED [ml]	164.61 (73.90)	160.55 (70.56)	<0.001	0.99
	LVV ES [ml]	99.06 (78.36)	96.40 (76.19)	0.001	1.00
	RVV ED [ml]	152.99 (54.88)	149.80 (53.88)	0.024	0.97
	RVV ES [ml]	85.88 (52.20)	87.37 (53.36)	0.310	0.96
	LVM [g]	129.96 (50.75)	128.83 (50.96)	0.180	0.99
	LVEF [%]	46.38 (19.82)	46.41 (20.51)	0.959	0.97
	RVEF [%]	46.73 (17.80)	44.86 (18.65)	0.043	0.88
SSC	LVV ED [ml]	163.52 (43.19)	162.51 (41.72)	0.305	0.98
	LVV ES [ml]	71.15 (42.48)	73.13 (43.68)	0.017	0.98
	RVV ED [ml]	148.71 (32.02)	142.65 (32.94)	0.001	0.87
	RVV ES [ml]	60.75 (20.17)	62.97 (21.44)	0.014	0.92
	LVM [g]	138.95 (36.98)	138.35 (34.58)	0.662	0.93
	LVEF [%]	58.80 (13.58)	57.32 (14.92)	0.002	0.95
	RVEF [%]	59.45 (8.65)	55.75 (11.43)	<0.001	0.82

Tabla 5.3: Valores medios y std de los indicadores de interés obtenidos de forma automática y manual por especialistas, junto con el valor *p-value* (p) y el coeficiente de *Pearson* (R).

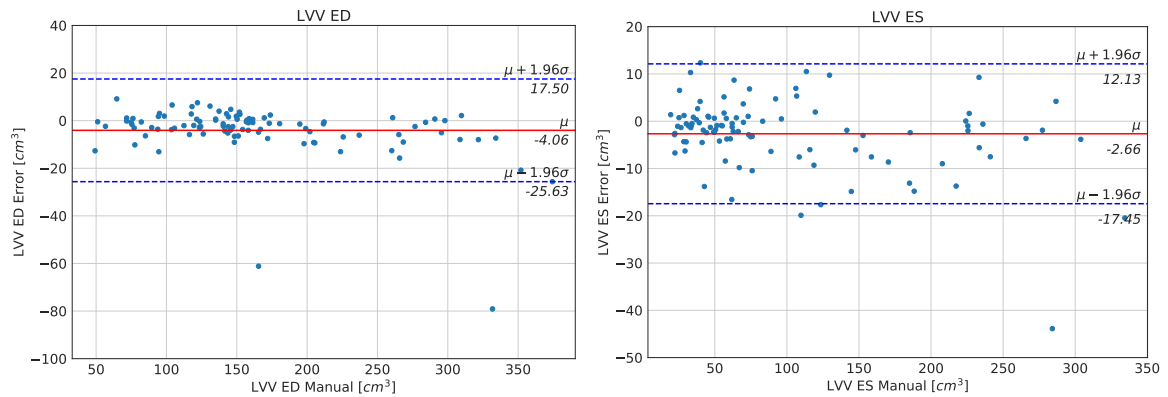


Figura 5.14: Gráficos de *Bland-Altman* del error cometido en la cuantificación automática de los indicadores LVV ED y LVV ES de los datos públicos (ACDC).

Como puede observarse (Figs. 5.14 y 5.15), la mayor cantidad de los errores quedan contenidos en el intervalo de confianza en los cuatro casos. A su vez, son comparables al error inter-observador que figura en la bibliografía^[36]. Allí se reportan valores de -2.2 ± 4.7 ml para LVV ED y -2.4 ± 4.7 ml para LVV ES, siendo de -4.06 ± 11.00 y -2.66 ± 7.54 respectivamente, para la cuantificación automática (Fig. 5.14). Un análisis similar pudo observarse para el resto de mediciones, por ejemplo se observó un error para LVEF de 0.02 ± 4.6 [%] (ACDC) y -1.48 ± 4.51 [%] (SSC), siendo de 2.7 ± 6.6 [%] en la bibliografía.

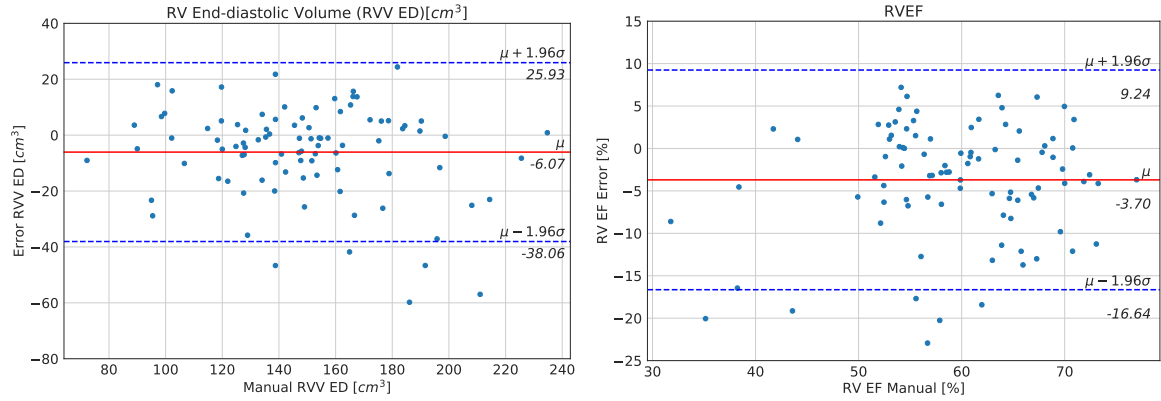


Figura 5.15: Gráficos de *Bland-Altman* del error cometido en la cuantificación automática de los indicadores RVV ED y RVEF de los datos regionales (SSC).

5.4. Robustez de la cuantificación cardiaca

Como se observó en la sección anterior, las técnicas de aprendizaje profundo demuestran un gran potencial para identificar diversas estructuras en imágenes cardiovasculares. Más aún, publicaciones recientes^[37,38] de la *American College of Cardiology* y *Frontiers in Cardiovascular Medicine* enumeran los prometedores logros de la inteligencia artificial en este campo. Sin embargo, estas estrategias aún presentan inconvenientes respecto a su robustez y reproducibilidad^[39], lo que hace que su implementación en la práctica clínica muchas veces no sea posible. Surge entonces la necesidad de comprender de forma objetiva las ventajas y debilidades a la hora de trasladar estos métodos al ámbito clínico. En este sentido, se evalúa a continuación la robustez de los modelos propuestos al ser evaluados sobre bases de datos diferentes a las de su entrenamiento. A partir de aquí, al referirse a los casos ACDC o SSC, se hará referencia a los modelos entrenados y evaluados sobre una misma BD.

Al estudiar los modelos entrenados con datos de ACDC y evaluados sobre SSC se observó que la precisión disminuye considerablemente respecto a cuando fueron entrenados y validados con los datos regionales (SSC). Frente a este comportamiento, se realizó un experimento adicional en el que se entrenaron los modelos de redes neuronales propuestos con los datos de ACDC y SSC juntos (ACDC+SSC). Usando nuevamente la estrategia de validación cruzada con 5 folds, se evaluaron dichos modelos y se calcularon los indicadores para los datos de SSC (Tabla 5.4).

En este experimento puede verse que el desempeño disminuye notablemente cuando se evalúan los modelos entrenados con los datos de ACDC sobre SSC. En particular, el error cometido al estimar la LVEF aumenta a un valor de $-3.20 \pm 18.85[\%]$, superior al error inter observador presentado anteriormente ($2.7 \pm 6.6[\%]$) y con una desviación considerablemente mayor, lo cual no es aceptable para la práctica clínica. Por otro lado, los modelos entrenados con ACDC+SSC y evaluados sobre SSC no muestran esta disminución en su desempeño. Más aún, los errores cometidos son comparables y se observan algunas mejoras en los ajustes respecto al caso SSC. Particularmente, se observó un incremento en el coeficiente de Pearson: RVV ED (0.90 vs 0.87), RVV ES (0.93 vs 0.92) y LVM (0.95 vs 0.93); aunque éste disminuye dos puntos en LVV ED (0.96 vs 0.98) y un punto en LVEF (0.94 vs 0.95).

		Manual	CardIAC	p	R	Bland-Altman
Ent: ACDC Test: SSC	LVV ED [ml]	163.52 (43.19)	134.84 (39.31)	<0.001	0.58	-28.68 (-102.89 a 45.53)
	LVV ES [ml]	71.15 (42.48)	60.21 (39.26)	<0.001	0.90	-10.94 (-46.71 a 24.82)
	RVV ED [ml]	148.71 (32.02)	129.53 (44.64)	<0.001	0.59	-19.18 (-90.34 a 51.98)
	RVV ES [ml]	60.75 (20.17)	61.62 (27.54)	0.664	0.72	0.87 (-36.52 a 38.26)
	LVM [g]	138.95 (36.98)	98.75 (31.11)	<0.001	0.59	-40.20 (-101.28 a 20.88)
	LVEF [%]	58.80 (13.58)	55.61 (27.10)	0.109	0.77	-3.20 (-40.14 a 33.74)
	RVEF [%]	59.45 (8.65)	50.46 (21.52)	<0.001	0.42	-8.99 (-47.20 a 29.23)
Ent: ACDC+SSC Test: SSC	LVV ED [ml]	163.52 (43.19)	161.21 (41.11)	0.064	0.96	-2.30 (-25.22 a 20.62)
	LVV ES [ml]	71.15 (42.48)	72.19 (43.25)	0.267	0.98	1.04 (-16.38 a 18.46)
	RVV ED [ml]	148.71 (32.02)	144.44 (33.05)	0.007	0.90	-4.27 (-33.06 a 24.51)
	RVV ES [ml]	60.75 (20.17)	63.90 (23.03)	0.001	0.93	3.15 (-13.93 a 20.24)
	LVM [g]	138.95 (36.98)	139.98 (34.73)	0.413	0.95	1.03 (-22.37 a 24.43)
	LVEF [%]	58.80 (13.58)	57.49 (15.22)	0.017	0.94	-1.31 (8.82 a -11.45)
	RVEF [%]	59.45 (8.65)	55.87 (11.70)	<0.001	0.82	-3.58 (-16.74 a 9.58)

Tabla 5.4: Valores medios de las variables bajo estudio, obtenidas de forma automática y manual por especialistas, junto con el valor *p-value* (p), el coeficiente de *Pearson* (R) y el error medio con los intervalos de confianza ± 1.96 std del estudio de Bland-Altman.

Discusiones

El análisis de la robustez del método al evaluar los modelos entrenados con una BD pública sobre una BD clínica regional deja en evidencia las limitaciones del enfoque automático. Este fenómeno es un indicio de la dificultad de generalización que presentan los distintos algoritmos de *machine learning*, particularmente cuando se trata de pocos datos, como es el caso de estudio. La causa principal de este comportamiento se debe a las diferencias que existen entre los datos públicos (ACDC) y los regionales (SSC). Los datos públicos en general están pensados para competencias y entrenamiento de algoritmos, donde normalmente las imágenes cuentan con una calidad considerablemente mejor que las imágenes que se obtienen en el trabajo del día a día en la clínica. En cambio, estas últimas suelen contener diversos artefactos, mayor variabilidad en su calidad e incluso contemplan la delineación en los casos que el miocardio y el RV no se encuentran completos (Fig. 5.9), las cuales generalmente se descartan en las BD públicas.

Al analizar los modelos entrenados con los datos ACDC+SSC y evaluados con SSC, se pudo observar que los indicadores derivados de las segmentaciones presentaban una precisión comparable a la obtenida por especialistas y errores comparables a los casos en que se entrenó y evaluó sobre SSC. La práctica de combinar bases cuyos datos son representados por distribuciones de probabilidad diferentes, como se pudo verificar para ACDC y SSC, podría ser un camino que ayude a mejorar la capacidad de generalización de los modelos. Por ejemplo, en el trabajo de *Ranftl et. al.*^[40], proponen y demuestran técnicas de mezclado de bases de datos diferentes que permiten mejorar la capacidad de generalización de los modelos entrenados. Por otro lado, en el trabajo de *C. Chen et. al.*^[38] se enfatiza que uno de los desafíos a superar en la detección de estructuras cardiovasculares con estas técnicas es la capacidad de generalización a nuevos datos que presenten distintas patologías y hayan sido obtenidos con otros equipos, lo cual podría lograrse al incorporar casos de distintas bases de datos en estudios multicéntricos. Esta motivación se ve reflejada en proyectos de generación de bases de datos de gran magnitud como es el caso de *Biobank*^[41].

Capítulo 6

Conclusiones y trabajos futuros

En este trabajo interdisciplinario de ingeniería se pudo diseñar y desarrollar CardIAC, una aplicación de código abierto que permite identificar estructuras cardiovasculares y cuantificar la función cardíaca de forma automática en estudios SAX MR-C, en tiempos compatibles con los de la clínica y con errores dentro del rango inter-observador actual. Hasta donde se tiene conocimiento, no hay en el país otra herramienta como la que se propone, sencilla de utilizar y basada en técnicas de aprendizaje profundo, que pueda ajustarse a las necesidades de diferentes centros clínicos nacionales para obtener cuantificaciones precisas en segundos o pocos minutos (~ 30 segundos en GPU y ~ 2 minutos en CPU).

Para lograr la identificación de estructuras cardíacas de interés, se utilizaron dos redes neuronales profundas: una red U-Net previamente entrenada para identificar el LV y un nuevo modelo basado en la red U-Net que incorpora información anatómica mediante el uso de un modelo generativo, constituido por un *autoencoder variacional*. A raíz del trabajo colaborativo con el equipo del Laboratorio de Procesamiento de Imágenes y Señales del Departamento de Física Médica (Instituto Balseiro), fue posible desarrollar dicho modelo, el cual se denominó GUVU-Net.

Mediante la primera red neuronal se logró construir una ROI que abarque la totalidad del corazón, aún en los casos de mayor error de detección del centro del LV (20 píxeles (31.15 mm)). Por otro lado, con el modelo GUVU-Net fue posible identificar las estructuras cardíacas y cuantificar los indicadores de interés (LVV, RVV, LVM, LVEF y RVEF) con errores dentro del rango inter observador cometido por los especialistas. En particular, en la cuantificación de la LVEF se observaron los errores $0.02 \pm 4.6\%$ (ACDC) y $-1.48 \pm 4.51\%$ (SSC), siendo de $2.7 \pm 6.6\%$ en la bibliografía. Sin embargo, aún es necesario realizar refinamientos, especialmente para mejorar la precisión en la detección del RV. Para ello, podría ser necesario un ajuste de hiperparámetros del modelo propuesto.

Se realizó un estudio cualitativo de la influencia del *prior* presente en el modelo GUVU-Net, que sugiere que la información adquirida en su entrenamiento permite disminuir el número de segmentaciones espurias y anatómicamente no plausibles. Sin embargo, es necesario un análisis más profundo que permita dilucidar el verdadero aporte de esta información al modelo, como podría ser el estudio de la información codificada en el espacio latente de ambas arquitecturas y analizar otras parametrizaciones

de la información en dicho espacio que sean diferentes a la normal, entre otras cosas. Si bien el número de segmentaciones parece reducirse, aún no logran ser eliminadas por completo. Asimismo, es posible que otros métodos de restricción de los modelos ayuden a mejorar aún más la generación de estructuras anatómicamente plausibles. Por ejemplo, los métodos actuales de post-procesado mediante el uso de VAEs han demostrado resultados muy prometedores que se sugiere analizar.

Si bien las técnicas de aprendizaje profundo presentan gran potencial en la identificación de estructuras cardíacas, aún presentan inconvenientes respecto a su robustez y reproducibilidad, lo que puede imposibilitar su implementación directa en la práctica clínica. Esto marca la necesidad de comprender de forma objetiva las ventajas y debilidades a la hora de trasladar estos métodos al ámbito clínico. Es por ello que se analizó la robustez del método propuesto al evaluar los modelos entrenados con la base de datos pública (ACDC) sobre los datos regionales (SSC). Este experimento dejó en evidencia las limitaciones del enfoque automático, ya que se pudo observar una disminución considerable en la precisión al cuantificar los indicadores de interés. Este fenómeno es un indicio de la dificultad de generalización que presentan los distintos algoritmos de *machine learning*, particularmente cuando se trata de pocos datos, como es el caso de las imágenes cardíacas de resonancia magnética. Sin embargo, al considerar un modelo entrenado sobre todo el conjunto de datos de ACDC+SSC, se logró recuperar el buen desempeño obtenido anteriormente. Esto demuestra los beneficios de incorporar un mayor número de datos con diferentes características (adquisición, patologías presentes, etc), lo cual es consistente con proyectos actuales de generación de datos clínicos confiables, objetivos y con diversas características de adquisición, como es el caso de *Biobank*.

En este sentido, un trabajo interdisciplinario parece ser el camino hacia la generación de herramientas y técnicas que permitan facilitar aún más los enfoques disponibles para la obtención de diagnósticos precisos, rápidos y objetivos en la clínica, con la capacidad de adaptarse a las nuevas tecnologías y necesidades actuales. Bajo esta premisa, CardIAc se diseñó con la posibilidad de incorporar modelos alternativos al propuesto para identificar el LV, RV y Myo, lo cual la convierte en una herramienta versátil y capaz de adecuarse al contexto regional en la que sea utilizada.

Bibliografía

- [1] H. Ritchie, “Causes of death,” *Our World in Data*, 2018. <https://ourworldindata.org/causes-of-death>.
- [2] World Health Organization. Office of Library and Health Literature Services, “Cardiovascular diseases (cvds),” 2017. [https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-\(cvds\)](https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds)).
- [3] M. Cikes and S. D. Solomon, “Beyond ejection fraction: an integrative approach for assessment of cardiac structure and function in heart failure,” *European Heart Journal*, vol. 37, no. 21, p. 1642–1650, 2015.
- [4] J. Román, J. Candell-Riera, R. Arnold, P. Sanchez, S. Agudé-Bruix, J. Bermejo, A. Revilla, A. Villa, H. Cuellar, C. Hernandez, and F. Fernández-Avilés, “Análisis cuantitativo de la función ventricular izquierda como herramienta para la investigación clínica. fundamentos y metodología,” *Revista Española De Cardiología*, vol. 62, pp. 535–551, 05 2009.
- [5] T. H. Marwick, “Ejection fraction pros and cons,” *Journal of the American College of Cardiology*, vol. 72, no. 19, p. 2360–2379, 2018.
- [6] S. E. Petersen, N. Aung, M. M. Sanghvi, F. Zemrak, K. Fung, J. M. Paiva, J. M. Francis, M. Y. Khanji, E. Lukaschuk, A. M. Lee, and et al., “Reference ranges for cardiac structure and function using cardiovascular magnetic resonance (cmr) in caucasians from the uk biobank population cohort,” *Journal of Cardiovascular Magnetic Resonance*, vol. 19, no. 1, 2017.
- [7] A. R. Patel and C. M. Kramer, “Role of cardiac magnetic resonance in the diagnosis and prognosis of nonischemic cardiomyopathy,” *Journal of the American College of Cardiology: Cardiovascular Imaging*, vol. 10, 10 2017.
- [8] A. Rosset, L. Spadola, and O. Ratib, “Osirix: An open-source software for navigating in multidimensional dicom images,” *Journal of Digital Imaging*, vol. 17, no. 3, p. 205–216, 2004.
- [9] E. Heiberg, J. Sjögren, M. Ugander, M. Carlsson, H. Engblom, and H. Arheden, “Design and validation of segment - freely available software for cardiovascular image analysis,” *BMC Medical Imaging*, vol. 10, no. 1, 2010.
- [10] W. Bai, M. Sinclair, G. Tarroni, O. Oktay, M. Rajchl, G. Vaillant, A. M. Lee, N. Aung, E. Lukaschuk, M. M. Sanghvi, and et al., “Automated cardiovascular magnetic resonance image analysis with fully convolutional networks,” *Journal of Cardiovascular Magnetic Resonance*, vol. 20, no. 1, 2018.

- [11] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” *Lecture Notes in Computer Science Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, p. 234–241, 2015.
- [12] A. H. Curiale, F. D. Colavecchia, P. Kaluza, R. A. Isoardi, and G. Mato, “Automatic myocardial segmentation by using a deep learning network in cardiac mri,” in *2017 XLIII Latin American Computer Conference (CLEI)*, pp. 1–6, Sept. 2017.
- [13] A. Curiale, F. Colavecchia, and G. Mato, “Automatic quantification of the lv function and mass: A deep learning approach for cardiovascular mri,” *Computer Methods and Programs in Biomedicine*, vol. 169, pp. 37–50, 02 2019.
- [14] “Human anatomy planes.” <https://www.anatomynote.com/human-anatomy/gross-view-of-human-body/human-anatomy-planes/>. Accessed: 13-10-2020.
- [15] C. Mitchell, P. S. Rahko, L. A. Blauwet, B. Canaday, J. A. Finstuen, M. C. Foster, K. Horton, K. O. Ogunyankin, R. A. Palma, E. J. Velazquez, and et al., “Guidelines for performing a comprehensive transthoracic echocardiographic examination in adults: Recommendations from the american society of echocardiography,” *Journal of the American Society of Echocardiography*, vol. 32, no. 1, p. 1–64, 2019.
- [16] “Diagram of the human heart.” <https://commons.wikimedia.org/w/index.php?curid=11429447>. Accessed: 13-10-2020.
- [17] G. J. Tortora and B. H. Derrickson, *Principles of Anatomy and Physiology*. Wiley, 12 ed., 2008.
- [18] A. Frangi, W. Niessen, and M. Viergever, “Three-dimensional modeling for functional analysis of cardiac images: A review,” *IEEE transactions on medical imaging*, vol. 20, pp. 2–25, 02 2001.
- [19] S. E. Petersen, M. Y. Khanji, S. Plein, P. Lancellotti, and C. Bucciarelli-Ducci, “European association of cardiovascular imaging expert consensus paper: a comprehensive review of cardiovascular magnetic resonance normal values of cardiac chamber size and aortic root in adults and recommendations for grading severity,” *European Heart Journal - Cardiovascular Imaging*, 09 2019.
- [20] A. G. Gheorghe, A. Fuchs, C. Jacobsen, K. F. Kofoed, R. Møgelvang, N. Lynnerup, and J. Banner, “Cardiac left ventricular myocardial tissue density, evaluated by computed tomography and autopsy,” *BMC Medical Imaging*, vol. 19, 12 2019.
- [21] R. Kline, “Cybernetics, automata studies, and the dartmouth conference on artificial intelligence,” *IEEE Annals of the History of Computing*, vol. 33, no. 4, p. 5–16, 2011.
- [22] A. Kaplan and M. Haenlein, “Siri, siri, in my hand: Who’s the fairest in the land? on the interpretations, illustrations, and implications of artificial intelligence,” *Business Horizons*, vol. 62, no. 1, pp. 15 – 25, 2019.
- [23] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2017.

- [24] Y. Lecun, P. Haffner, and Y. Bengio, “Object recognition with gradient-based learning,” 08 2000.
- [25] Bai Kunlun, “A comprehensive introduction to different types of convolutions in deep learning.” <https://towardsdatascience.com/a-comprehensive-introduction-to-different-types-of-convolutions-in-deep-learning-669281e58215>.
- [26] D. P. Kingma and M. Welling, “Auto-Encoding Variational Bayes,” in *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.
- [27] W. J. Schroeder and K. M. Martin, “The visualization toolkit,” *Visualization Handbook*, p. 593–614, 2005.
- [28] H. J. Johnson, M. M. McCormick, and L. Ibanez, *Template: The ITK Software Guide Book 1: Introduction and Development Guidelines-Volume 1*, 2015.
- [29] A. Bernardo, “Estimación del esfuerzo miocárdico a partir del procesamiento de imágenes médicas,” Maestría en Física Médica, Master’s thesis, Instituto Balseiro, Diciembre 2020.
- [30] A. Bernardo, L. Dellazoppa, G. Mato, and A. H. Curiale, *Manual de uso: CardIAC*, 2020. <https://cutt.ly/cardIAC-manual>.
- [31] C. C. Imaging, “cvi42.” <https://www.circlecvi.com/>.
- [32] O. Bernard, A. Lalande, C. Zotti, F. Cervenansky, X. Yang, P.-A. Heng, I. Cetin, K. Lekadir, O. Camara, M. A. G. Ballester, and et al., “Deep learning techniques for automatic mri cardiac multi-structures segmentation and diagnosis: Is the problem solved?,” *IEEE Transactions on Medical Imaging*, vol. 37, no. 11, p. 2514–2525, 2018.
- [33] A. H. Curiale, L. Dellazoppa, and G. Mato, “Cardiac: Cuantificación de la función cardíaca mediante la detección automática del ventrículo izquierdo en imágenes de resonancia magnética nuclear,” in *Taller Argentino de Computación Científica*, (San Luis, Argentina), Facultad de Ciencias Físico Matemáticas y Naturales, Universidad Nacional de San Luis, 2019.
- [34] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.
- [35] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” *CoRR*, vol. abs/1502.03167, 2015.
- [36] A. Suinesiaputra, M. M. Sanghvi, N. Aung, J. M. Paiva, F. Zemrak, K. Fung, E. Lukaschuk, A. M. Lee, V. Carapella, Y. J. Kim, and et al., “Fully-automated left ventricular mass and volume mri analysis in the uk biobank population cohort: evaluation of initial results,” *The International Journal of Cardiovascular Imaging*, vol. 34, no. 2, p. 281–291, 2017.

- [37] D. Dey, P. J. Slomka, P. Leeson, D. Comaniciu, S. Shrestha, P. P. Sengupta, and T. H. Marwick, “Artificial intelligence in cardiovascular imaging,” *Journal of the American College of Cardiology*, vol. 73, no. 11, pp. 1317–1335, 2019.
- [38] C. Chen, C. Qin, H. Qiu, G. Tarroni, J. Duan, W. Bai, and D. Rueckert, “Deep learning for cardiac image segmentation: A review,” *Frontiers in Cardiovascular Medicine*, vol. 7, 2020.
- [39] Q. Zheng, H. Delingette, N. Duchateau, and N. Ayache, “3d consistent & robust segmentation of cardiac images by deep learning with spatial propagation,” *CoRR*, vol. abs/1804.09400, 2018.
- [40] R. Ranftl, K. Lasinger, D. Hafner, K. Schindler, and V. Koltun, “Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, p. 1–1, 2020.
- [41] C. Sudlow, J. Gallacher, N. Allen, V. Beral, P. Burton, J. Danesh, P. Downey, P. Elliott, J. Green, M. Landray, and et al., “Uk biobank: An open access resource for identifying the causes of a wide range of complex diseases of middle and old age,” *PLOS Medicine*, vol. 12, no. 3, 2015.